

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО»**

**ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ  
КАФЕДРА ПРОМИСЛОВОЇ ЕЛЕКТРОНІКИ**

«На правах рукопису»  
УДК 004.942:519.876

«До захисту допущено»  
Завідувач кафедри

\_\_\_\_\_ Ю.С. Ямненко  
(підпис) (ініціали, прізвище)

“\_\_” \_\_\_\_\_ грудня 2019 р.

**Магістерська дисертація  
на здобуття ступеня магістра**

зі спеціальності 171 Електроніка  
(код і назва)

освітня програма (спеціалізація) Електронні компоненти і системи

на тему: Прогнозування генерації електроенергії в MicroGrid на базі  
нейронної мережі

Виконав (-ла): студент (-ка) II курсу, групи ДС-81мп  
(шифр групи)

\_\_\_\_\_ Живогляд Олег Станіславович \_\_\_\_\_  
(прізвище, ім'я, по батькові) (підпис)

Науковий керівник зав. каф., професор, д.т.н., Ямненко Ю.С. \_\_\_\_\_  
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант \_\_\_\_\_  
(назва розділу) (науковий ступінь, вчене звання, прізвище, ініціали) (підпис)

Рецензент \_\_\_\_\_  
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Консультант  
по нормоконтролю доцент, к.т.н., Батрак Л.М. \_\_\_\_\_  
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській дисертації  
немає запозичень з праць інших авторів без  
відповідних посилань.

Студент \_\_\_\_\_  
(підпис)

Київ – 2019 року

**Національний технічний університет України  
“Київський політехнічний інститут  
імені Ігоря Сікорського”**

Факультет електроніки  
( повна назва )

Кафедра промислової електроніки  
( повна назва )

Рівень вищої освіти – другий (магістерський) за освітньо - професійною програмою

Спеціальність 171 Електроніка  
(шифр і назва)

Освітня програма (спеціалізація) Електронні компоненти і системи

**ЗАТВЕРДЖУЮ**  
Завідувач кафедри

\_\_\_\_\_  
( підпис)                      Ю.С. Ямненко  
( прізвище ініціали )

«06» листопада 2019 року

**З А В Д А Н Н Я**

**НА МАГІСТЕРСЬКУ ДИСЕРТАЦІЮ СТУДЕНТУ**

Живогляд Олег Станіславович

(прізвище, ім'я, по батькові)

1. Тема дисертації Прогнозування генерації електроенергії в MicroGrid на базі нейронної мережі

науковий керівник дисертації Ямненко Юлія Сергіївна, професор, д.т.н.,  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від « 07 » листопада 2019 року № 3859-с

2. Термін подання студентом дисертації 10.12.2019

3. Об'єкт дослідження: процеси генерації електричної енергії відновлювальними та альтернативними джерелами у складі систем розподіленої генерації MicroGrid

4. Вихідні дані: прогнозування генерації електричної енергії у MicroGrid з використанням математичного апарату штучних нейронних мереж

5. Перелік завдань, які потрібно розробити: 1.Огляд існуючих структур систем розподіленої генерації MicroGrid, аналіз їх характерних особливостей. 2.Порівняльний аналіз методів прогнозування, вибір та обґрунтування методу штучних нейронних мереж для поставленої задачі прогнозування у MicroGrid. 3.Дослідження типів та структур штучних нейронних мереж, їх порівняння та вибір найбільш доцільної структури. 4.Розробка способу прогнозування електрогенерації на базі штучної нейронної мережі. 5.Моделювання роботи штучної нейронної мережі, обробка та аналіз результатів.

6. Орієнтовний перелік графічного (ілюстративного) матеріалу Слайди презентації

7. Орієнтовний перелік публікацій: 1.Живогляд О.С. Прогнозування електроспоживання на базі нейронної мережі. // Електронна та Акустична Інженерія – 2019. Том 2, №1. – С. 39-42. ISSN 2524-2725. 2.Zhyvoglyad O.S. System of automatic control of power usage based on a neural network» - стаття подана до публікації у фаховому періодичному виданні «Світлотехніка та електроенергетика» пройшла рецензування 18.11.2019 та прийнята до друку.

#### 8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

9. Дата видачі завдання 06.10.2019 року

### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
<b>1</b>	Огляд структур, характеристик та сфер застосування MicroGrid	02.09.2019-16.09.2019	Виконано
<b>2</b>	Дослідження способів застосування нейронних мереж у MicroGrid	16.09.2019-30.09.2019	Виконано
<b>3</b>	Дослідження типів та структур штучних нейронних мереж, їх порівняння та вибір найбільш доцільної структури	01.10.2019-14.10.2019	Виконано
<b>4</b>	Розробка способу прогнозування електрогенерації на базі штучної нейронної мережі.	14.10.2019-31.10.2019	Виконано
<b>5</b>	Моделювання роботи штучної нейронної мережі, обробка та аналіз результатів.	01.11.2019-24.11.2019	Виконано
<b>6</b>	Розробка стартап-проекту	24.11.2019-08.12.2019	Виконано

Студент

\_\_\_\_\_  
(підпис)

Живогляд О.С.  
(ініціали, прізвище)

Науковий керівник дисертації

\_\_\_\_\_  
(підпис)

Ямненко Ю.С.  
(ініціали, прізвище)

## АНОТАЦІЯ

У магістерській дисертації було розроблено метод прогнозування обсягів генерації від альтернативних джерел електроенергії у системі MicroGrid на базі штучних нейронних мереж. Було вибрано необхідний для поставленої задачі метод прогнозування і підібрано для його реалізації 2 моделі нейронних мереж: «Умовно обмежену машину Больцмана» (англ. CRBM) та «Нелінійну авторегресивну модель з екзогенним входом» (англ. NARX).

В результаті моделювання було отримано такі висновки, що «Нелінійна авторегресивна модель з екзогенним входом» краще підходить для поставлених задач. Також було розроблено стартап-проект по використанню данної системи.

Ключові слова: *Micro Grid, нейронна мережа, CRBM, контрастна дивергенція, NARX, оптимізація Левенберга-Маркарда.*

## АННОТАЦИЯ

В магистерской диссертации был разработан метод прогнозирования объемов генерации от альтернативных источников электроэнергии в системе MicroGrid на базе искусственных нейронных сетей.

Было выбрано необходимый для поставленной задачи метод прогнозирования и подобрано для его реализации 2 модели нейронных сетей: «Условно ограниченную машину Больцмана» (англ. CRBM) и «Нелинейную авторегресивную модель с экзогенным входом» (англ. NARX).

В результате моделирования были получены следующие выводы, «Нелинейная авторегресивная модель с экзогенным входом» лучше подходит для поставленных задач. Также был разработан стартап-проект по использованию данной системы.

Ключевые слова: *Micro Grid, нейронная сеть, CRBM, контрастная дивергенция, NARX, оптимизация Левенберга-Маркарда.*

## ANNOTATION

In the master's thesis was developed method of forecasting the generation volumes from alternative sources of electricity in the MicroGrid system based on artificial neural networks.

The prediction method required for the task was selected its 2 neural network models implementation: “ Conditional restricted Boltzmann machine ” (CRBM) and “ Nonlinear Autoregressive model process with eXogenous input ” (NARX).

As a result of the simulation, it was concluded that "Nonlinear autoregressive model with exogenous input" is better suited for the tasks. A startup project for using this system was also developed.

Keywords: Micro Grid, neural network, CRBM, contrast divergence, NARX, Levenberg-Marquard optimization.

## ЗМІСТ

ВСТУП.....	4
РОЗДІЛ 1. ПРИНЦИПИ ПРОГНОЗНОГО КЕРУВАННЯ У MICROGRID.....	7
1.1. Система розподіленої генерації MicroGrid та її характерні особливості.....	7
1.2. Основні поняття теорії штучних нейронних мереж у застосуванні до задач прогнозування .....	14
1.3. Прогнозування генерації в MicroGrid на базі нейронних мереж .....	24
1.4. Основні моделі нейронних мереж.....	32
Висновки до розділу 1.....	40
РОЗДІЛ 2 МОДУЛІ ПРОГНОЗУВАННЯ НА БАЗІ НЕЙРОННИХ МЕРЕЖ.....	41
2.1. Рекурентна нейронна мережа RBM.....	41
2.2. Рекурентна нейронна мережа NARX.....	46
Висновки до розділу 2.....	51
РОЗДІЛ 3 МОДЕЛЮВАННЯ НЕЙРОННИХ МЕРЕЖ.....	50
3.1. База даних часових рядів генерації електроенергії.....	50
3.2. Моделювання в MATLAB.....	54
3.3. Аналіз результатів моделювання.....	58
Висновки до розділу 3 .....	60
РОЗДІЛ 4 РОЗРОБКА СТАРТАП-ПРОЕКТУ.....	61
Висновки до розділу 4.....	65
ВИСНОВКИ.....	66
СПИСОК ЛІТЕРАТУРИ.....	67
ДОДАТКИ.....	71
РЕФЕРАТ.....	77

## ВСТУП

**Актуальність.** MicroGrid — складна електротехнічна система, що включає в себе різні підсистем, які об'єднані та взаємопов'язані одна з іншою. Ця мережа побудована з альтернативних та звичайних джерел енергії та різних навантажень, об'єднаних в одну систему. Головною особливістю MicroGrid є те, що, не беручи до уваги функціонування в рамках загальної мережі, довше вона ізольована через пониження вартості тарифу та найвищої енергоефективності. Системи, що живляться від альтернативних джерел є привабливими через те, що отримують дотації та знижки на розповсюдження енергії так званий «зелений тариф», а отже є економічно доцільними. Але альтернативні джерела генерують електроенергію нестабільно через погоду, денні цикли та інше, тому потребують більш складну систему керування.

При вирішенні цієї задачі постає питання прогнозування електрогенерації в окремих підсистемах. Враховуючи наявність великої кількості даних, що підлягають обробці у режимі реального часу, а також «людський фактор», що додатково ускладнює задачу, для прогнозування електрогенерації у MicroGrid, перспективним є застосування методу штучних нейронних мереж.

**Мета і завдання дослідження.** Метою магістерської дисертації є розробка методу прогнозування обсягів генерації від альтернативних джерел електроенергії у системі MicroGrid на базі штучних нейронних мереж.

Поставлена мета вимагає вирішення наступних наукових і практичних задач:

1. Огляд існуючих структур систем розподіленої генерації MicroGrid, аналіз їх характерних особливостей, формулювання та обґрунтування задач прогнозування генерації електроенергії від альтернативних джерел.
2. Порівняльний аналіз методів прогнозування, вибір та обґрунтування методу штучних нейронних мереж для поставленої задачі прогнозування у MicroGrid.

3. Дослідження типів та структур штучних нейронних мереж, їх порівняння та вибір найбільш доцільної структури для вирішення поставленої задачі.

4. Розробка способу прогнозування електрогенерації на базі штучної нейронної мережі.

5. Моделювання роботи штучної нейронної мережі, обробка та аналіз результатів.

**Об'єктом дослідження** є процеси генерації електричної енергії відновлювальними та альтернативними джерелами у складі систем розподіленої генерації MicroGrid.

**Предметом дослідження** є прогнозування генерації електричної енергії у MicroGrid з використанням математичного апарату штучних нейронних мереж.

**Зв'язок роботи з науковими програмами, темами, планами.** Науково-дослідна робота за темою магістерської дисертації виконувалась у відповідності до напрямків наукових досліджень та плану роботи кафедри промислової електроніки КПІ ім. Ігоря Сікорського.

**Методи дослідження.** Для вирішення задач прогнозування використано математичний апарат штучних нейронних мереж з архітектурою NARX. Для підтвердження теоретичних розрахунків використано програмне середовище MATLAB, в якому реалізовано запропоновану структуру нейронної мережі.

**Наукова новизна одержаних результатів** полягає в наступному:

- Вперше запропоновано використання математичного апарату штучних нейронних мереж для розв'язання задачі прогнозування генерації електричної енергії у MicroGrid;

- Проведено порівняльний аналіз різних структур штучних нейронних мереж, обгрунтовано вибір архітектури NARX для вирішення поставленої задачі.

**Практичне значення одержаних результатів:**

- застосування штучних нейронних мереж для прогнозування обсягів генерації електроенергії, що надходить від альтернативних джерел живлення у



MicroGrid, дозволяє оперувати з великими обсягами даних, що є надзвичайно актуально для сучасних інтелектуальних систем типу «розумний будинок», зокрема, з реалізацією концепції «Інтернету речей»;

- теоретичні результати роботи підтверджено результатами моделювання у програмному середовищі MATLAB, що свідчить про доцільність та обґрунтованість обраної архітектури для прогнозування.

**Апробація результатів роботи** здійснювалася в рамках наукової доповіді на XII міжнародній науково-технічній конференції молодих вчених «Електроніка-2019», 23-25 квітня, 2019 р.

**Публікації.** Основні положення та результати магістерського дослідження висвітлено у наступних публікаціях:

1. Живогляд О.С. Прогнозування електроспоживання на базі нейронної мережі. // Електронна та Акустична Інженерія – 2019. Том 2, №1. – С. 39-42. ISSN 2524-2725.

2. Zhyvoglyad O.S. System of automatic control of power usage based on a neural network» - стаття подана до публікації у фаховому періодичному виданні «Світлотехніка та електроенергетика» <https://sds.kname.edu.ua/index.php/ru/kafedra/zhurnal-svitlotekhnika-ta-elektroenergetika>. Стаття пройшла рецензування 18.11.2019 р. та прийнята до друку.

**Структура та обсяг роботи.** Магістерська дисертація складається з 4 розділів, загальних висновків, списку літератури з 40 найменувань та 1 додатку. До кожного розділу є окремі висновки. Загальний обсяг роботи - 80 сторінок.

## **РОЗДІЛ 1. ПРИНЦИПИ ПРОГНОЗНОГО КЕРУВАННЯ У MICROGRID**

### **1.1. Система розподіленої генерації MicroGrid та її характерні особливості**

Основним драйвером для просування технології MicroGrid стала задача забезпечення енергоефективності. Насправді спочатку вирішенням вважалось пошук нових джерел енергозбереження. Ними можуть служити передача електроенергії на постійному струмі в магістральних мережах, застосування частотно-регульованих приводів, установка низьковитратних силових трансформаторів, використання населенням LED-технологій замість класичних ламп. В секторі економіки генеруючих потужностей інвестори активно підтримували підхід до відновлюваних джерел які б логічного доповнювали традиційне пальне - в основному вугілля, газ і ядерне паливо [1].

Якщо передбачити зростання цін на сировину і зменшення його запасів то виникають жорсткі вимоги щодо зниження витрат. Енергетика, не такий довгий час знаходиться в стабільному стані, вона пережила серію криз і виявилася не готова до нових ринкових умов, в яких з'явився ще один фактор - субсидії на відновлювальні джерела енергії (ВДЕ) [1].

Завдяки субсидіям на відновлювальні джерела енергії на ринку з'являється багато нових гравців, і використання ВДЕ стало дуже популярно. Але так як реальна інфраструктура не була готова до оновлення, більшість нових джерел не змогли підключатися до мереж, які нездатні до роботи з ВДЕ. У зв'язку з технічними особливостями застосування нових джерел було потрібно і кращий досвід експлуатації відновлювальних джерел. Потужності таких джерел обчислювалися в більшості в 1 або 10 МегаВат, що також негативно впливало на мережу в цілому [1]. У зв'язку з цим в кожній подібній події потрібен був власний активний резерв, що реагував би на флуктуації в роботі відновлювальних джерел електроенергії. Так як у об'єднаних мережевих системах такі завдання можуть бути виконані, то в ізолюваних мережах (тих, які не підключені до магістральної

мережі або в яких вона підтримується на низькому рівні) це є неможливим і не реальним або якщо і можливо то тільки за рахунок видачі електроенергії поганої якості. Тоді ж як, головні якісні характеристики саме для споживачів - це незмінна напруга і частота. Тому рішення цих проблем, пов'язаних з роботою традиційних і відновлювальних джерел в ізольованих мережах, є на сьогодні дуже важливим питанням в енергетичній галузі [2].

MicroGrid — складна електротехнічна система, що включає в себе різні підсистем, які об'єднані та взаємопов'язані одна з іншою. Ця мережа побудована з альтернативних та звичайних джерел енергії та різних навантажень, об'єднаних в одну систему. Головною особливістю MicroGrid є те, що, не беручи до уваги функціонування в рамках загальної мережі, довше вона ізольована через пониження вартості тарифу та найвищої енергоефективності. Генеруючі об'єкти це комбінація звичайних джерел, що працюють на дизелі або газі, міні-гідроелектричні станції і ВДЕ (відновлювальні джерела енергії). Використовуються всі зазначені комбінації джерел, і якщо головною причиною для вибору ВДЕ є власне економічні субвенції (субсидії, зелений тариф), або економия витрат звичайного палива, то тип джерела електричної енергії великого змісту не має. Зверніть увагу, заміна звичайного джерела «зеленим» також є дуже позитивним показником в екологічному факторі. Якщо власнику мережі потрібно підвищити екологічність та поліпшити рівень життя споживачів, то для них використання ВДЕ ще і знижує оподаткування за рахунок зеленого тарифу. У розвинутих країнах цей аспект став вирішальним для застосування ВДЕ в локальних мережах таких як MicroGrid [2-3].

MicroGrid є головною системою для точного та ефективного підтримання взаємозв'язку в усіх наявних членах мережі, а також управління керуючим обладнанням для моніторингу. Система MicroGrid представляє масштабове рішення, котре могло б використовувати декілька сценарії при роботі з різним навантаженням від окремих будівник до цілих селищ. Мережа повинна бути оснащена особливими засобами зв'язку, що допомагають при контролі і управлінні усіма наявними електричними ресурсами даної мережі [2-3] (рис. 1.1).

Для контролю оптимізації в системі використовуються алгоритми динамічного управління, які контролюють ресурси. Кожен розподілений енергетичний ресурс контролюється і керується локально, але його робота координується через центральну систему керування. Алгоритми керування працюють в режимі реального часу для оптимізації навантаження мережі та забезпечення надійного енергопостачання енергії [3].

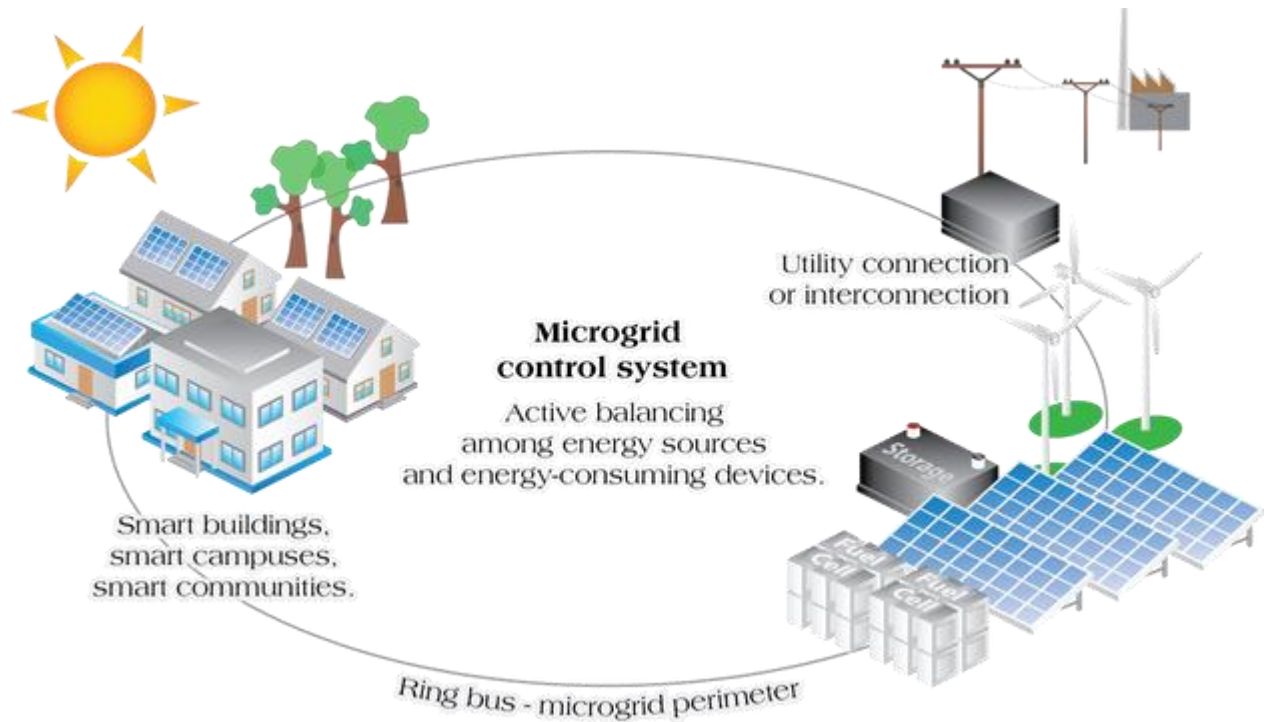


Рис. 1.1 Модель мережі MicroGrid

Однак структура MicroGrid, яка має надійні та переважачі характеристики, вона має і недоліки, які вимагають рішень для підтримки розгляду цієї концепції. Головні недоліки та проблематика MicroGrid [4,7]:

Малопотужна генерація – включає в себе такі елементи електроенергетичної системи (ЕЕС), що мають нові динамічні характеристики і можливості управління. Ось, наприклад, вітряні електроустановки (ВЕ) генерують енергію в змінному режимі використання, тоді якщо достатньо сумарних потужностей ВЕ створюють проблему при управлінні режимом регулювання частот ЕЕС. Якщо буде надто сильний вітер ВЕ припиняють

роботу, що при значних їх сумарних потужностях повинно впливати та є причиною значним обуреннями в ЕЕС, так що призводять до падіння стійкості мережі та є причинами аварії [4,7].

Значна можливість впливу малопотужної електрогенерації на генерацію вищих гармонік в системі. Спочатку, присутність малопотужних електрогенераторів призводить до падіння загального рівня. Потім, ряд малопотужних систем (ВЕ) підключається до загальної мережі, в яких генеруються в мережу вищі гармоніки [4,7].

Включення ВДЕ малопотужної електрогенерації до загальної мережі підвищує струми КЗ (кор. зам.), що може потребувати додаткових ліній зв'язку, заміни елементів запобіжників.

Малопотужна генерація підвищує складність диспетчерського керування ЕС, передаючи ці обов'язки систему керування. Проблематика в такому випадку спричинена досить високими невизначеностями робочими характеристиками малопотужних електрогенерацій через те що невраховано генерація та споживання, недостатність реальних даних про їх робочі режими [4,7].

Поява активних споживачів змінює ролі кінцевих споживачів у системі розподілу енергії від пасивного придбання електроенергії, до активного планування попиту та генерації відповідно до індивідуальних потреб. Це також робить можливим вивчення місцевих ВДЕ, а також переваги як гравців, так і системи. Однак новий сценарій приносить надзвичайну складність вивчення та керування. Виникає необхідність розробки нових підходів та інструментів для розуміння системи розподілу енергії, особливо для тих, хто приймає політичні рішення. Нова інтелектуальна мережа може відкрити нове бачення функціонування енергосистем і дозволить створити самоорганізовану спільноту активних споживачів з абсолютно новим підходом. Важливе значення в рамках злиття мають соціальні та технічні взаємодії, які впливають на глобальну продуктивність та потреби прагнути до макроконтролю власних зацікавлених мікро-гравців, які взаємодіють на соціальному та мережевому рівнях. Регулювання може бути здійснене за допомогою зовнішнього стимулювання як

з точки зору цінкових, так і соціальних цінностей. Іноді соціальні цінності можуть бути більш ефективними, ніж цінкові сигнали. Результати показують, що для максимізації ефективності політики слід ретельно розвивати відповідну соціальну культуру [6].

Активні споживачі надають інформацію про останній комфорт і користь та ставлення до них в кожному соціальному інтервалі часу в тому ж контрольному інтервалі часу в однаковому колі суспільства. В кожному контрольному інтервалі часу активні споживачі аналізують історичні ціни, щоб передбачити можливі коливання цін на споживання та генерацію. На основі нещодавно оновленої соціальної інформації та економічних сигналів від макро-гравців активні споживачі розраховують власну генерацію та попит, враховуючи ліміти генерації та попиту на наступний контрольний інтервал часу. Після самозабезпечення роль активного споживача визначається як продавець (генерація в надлишку) або покупець (дефіцит покоління). Під час участі в торгівлі в кожному контрольному інтервалі часу, споживач використовує стратегію, що не зменшує прибуток, щоб сформуванати кінцеву цінову пропозицію. У кожному контрольному інтервалі часу самоорганізовані переговори щодо цін і кількості серед активних споживачів в межах одного кола комунікації або на одній шині розглядають також балансувальні премії і циклічний заряд [4,6].

Агрегатор навантаження надає послуги від імені активних споживачів, бере участь у ринку місцевих громад на основі модифікованих вузлових цін, а також збирає інформацію про самоорганізовані кросшинні угоди та їхні відповідні коригувальні ставки для подання до ринку місцевих громад. Після отримання уточнених обсягів споживання та генерації та вузлових цін від ринку місцевих громад, агрегатор навантаження перераховує ціни на споживання та генерацію, щоб отримати плату за послугу. Потім уточнені обсяги та змінені ціни відправляються назад до відповідних посередників [4,6].

Локальний координатор електроенергії керує ринком місцевих громад, використовуючи котирування з агрегатора навантаження, малих традиційних

генераторів і навантажень в одній громаді, спрямовану на забезпечення найбільш ефективних стимулів, щоб спонукати поведінку учасників ринку до бажаного результату (наприклад, максимізацію соціального надлишку, мінімізація загальних збитків тощо) при дотриманні оперативних обмежень. Обмін енергією з іншими громадами також розглядається як одне з завдань локального координатора електроенергії. Оскільки локальний координатор електроенергії не може безпосередньо керувати попитом і генерацією активних споживачів, він використовує вузлові ціни як економічні стимули для прийняття рішень активних споживачів щодо їхнього попиту та генерації, що приводить всю систему до бажаного стану [5,8].

Загальне розуміння терміну MicroGrid в науково-дослідних роботах охоплює розробку нових стратегій управління потужністю та систем зв'язку для вирішення проблеми енергоефективності. Це пов'язано з розширенням використання відновлюваних джерел енергії та нових електричних навантажень (наприклад, теплових насосів та електричних транспортних засобів) [5,8].

Одним з новітніх підходів керування для розробки програмних додатків у галузі MicroGrid є мультиагентні системи керування (МАСК). МАСК складається з ряду окремих суб'єктів, званих агентами, де агенти взаємодіють між собою. Кожна структура може бути агентом. Наприклад, агент може бути пристроєм, комп'ютерною системою, програмним забезпеченням або організацією, що їх володіє. Кожен агент може як самостійно працювати для досягнення певної особистої мети, так і взаємодіяти з іншим агентом, щоб вирішити спільну мету [9]. Використання МАС в інжинірингу енергосистем стає досить поширеним явищем і представлене в літературі різними застосуваннями в різних архітектурах. Агенти - розсіяні компоненти розрахунку в енергосистемі. Вони здатні отримувати дані про напругу, струм та стан системи і діяти на приводи, щоб змінити стан енергосистеми. Як правило, три типи стратегій контролю використовуються для контролю агентів у МАСК. Це централізовані, децентралізовані та гібридні стратегії координації [8-10].

До переваг МАСК відносять [10]:

- розвиток механізму підлаштовування під умови, можливості покращення функціоналу і параметрів безпосередньо під час роботи;
- використання динамічного моделювання в реальному часі з передбаченням характеристик енергосистеми;
- обробка розподіленого інформаційно-комунікаційного набору даних, з повним взаємообміном і оптимізацією взаємодії між різними підсистемами, можливість інформаційного обміну не тільки даними, але й знаннями;
- застосування паралельних обчислень, які дозволяють різко підвищити швидкодію та на порядок економити обчислювальні ресурси;
- здатність здійснювати керування станом технічних комплексів і систем за допомогою оброблених підсистем технічних мереж і особливої здатності комп'ютеризованих технологій.

Можливості мультиагентних систем керування:

- модифікація структури і параметрів безпосередньо в процесі функціонування;
- застосування динамічного моделювання в реальному часі з прогнозуванням стану;
- організація розподіленого керування з оптимізацією взаємодії між підсистемами;
- можливість обміну як даними, так і знаннями; паралельні обчислення, що підвищують швидкість обчислень при економії обчислювальних ресурсів; керування станом технічних комплексів і систем на основі розподілених мережових інформаційних мереж і гнучкою інфраструктури обчислювальних компонентів [8-10].

Архітектура МАСК має містити:

- спеціалізовані бази знань (онтологій) для подання інформаційних потреб функціонування різних агентів;
- моделі процесів функціонально повної підсистеми спеціалізованих агентів, забезпечують пошук, уявлення, обробку та розподілення інформації;



- пристрій прогнозової генерації;
- пристрій інформаційно-технологічної взаємодії і сумісності агентів;
- підсистему формування керуючих впливів і реалізації керування;
- підсистему аналізу і планування [10].

Набула поширення архітектура МАСК на принципах штучного інтелекту, що використовує методи і засоби символного представлення даних. На сьогоднішня обмеженість архітектури – відсутність підсистеми моделювання для прогнозування своєї поведінки, поведінки інших агентів і зовнішнього середовища. Цю проблему можна вирішити саме за допомогою нейронної мережі.

## **1.2. Основні поняття теорії штучних нейронних мереж у застосуванні до задач прогнозування**

Наразі очевидно штучні нейронні мережі (ШНМ) — це новий тип популярних з научної та практичної точки зору обчислювальних структур. Ця популярність спровокована кількома успішними застосуваннями технології ШНМ, в сегментах, що раніше вважалися складними для реалізації на традиційних алгоритмах. Сама назва “нейронні мережі” використовується до всіх обчислювальних структур, що можуть нагадувати структуру мозку. Проте прогнозувати появу реального штучного інтелекту або хоч спрощеної копії, що наслідувала б роботу мозку, неварто [11].

Перша хвиля, що цікавилась нейронними мережами, також відома як коннекційні моделі або паралельно розподілена обробка з'явилася після впровадження спрощених нейронів Уоррена С. Мак-Каллока та Вальтера Піттса. Ці нейрони були представлені як моделі біологічних нейронів та як концептуальні компоненти для схем, які могли б виконувати обчислювальні завдання. Коли Мінський та Паперт опублікували свою книгу «Перцептрони» в якій вони показали недоліки моделі перцептрону, фінансування дослідів нейронної мережі було призупинено, а дослідники залишили спроби. Лише

кілька дослідників продовжували роботу з нейронною мережею, зокрема Теуво Кохонен, Стефан Гроссберг, Джеймс Андерсон та Куніхіко Фукусіма.

Інтерес до нейронних мереж з'явився лише після отримання важливих теоретичних результатів досягнутих на початку вісімдесятих, особливо це виявлення поширення помилок і нових розробок апаратних засобів, що збільшили обробну потужність. Це відновило інтерес у деякої кількості вчених, суми фінансування, кількість великих конференцій та кількість журналів, пов'язаних з нейронними мережами. У наш час більшість університетів мають нейронно мережеві групи в рамках кафедр інформатики, фізики, психології або біології.

Штучні нейронні мережі можна найбільш адекватно охарактеризувати як обчислювальні моделі з особливими властивостями, такими як здатність адаптуватися чи навчитися, узагальнювати або кластеризувати, впорядкувати дані та виконувати операції які ґрунтуються на паралельній обробці. Однак багато з перерахованих вище властивостей можна віднести до існуючих не нейронних моделей.

Інтригуюче питання полягає в тому, в якій мірі нейронний підхід виявляється більш підходящим для певних застосувань, ніж існуючі моделі. На сьогодні однозначної відповіді на це питання не знайдено.

Часто описуються паралелі з біологічними системами. Однак досі так мало відомо (навіть на найнижчому рівні клітин) щодо біологічних систем, які використовуються нами як приклади для створення моделей штучних нейронних мереж, що здається ми використовуємо надто спрощені біологічні моделі.

Штучна нейронна мережа складається з пулу простих одиниць обробки (нейронів) , які спілкуються посиленням сигналів один одному через велику кількість зважених з'єднань.

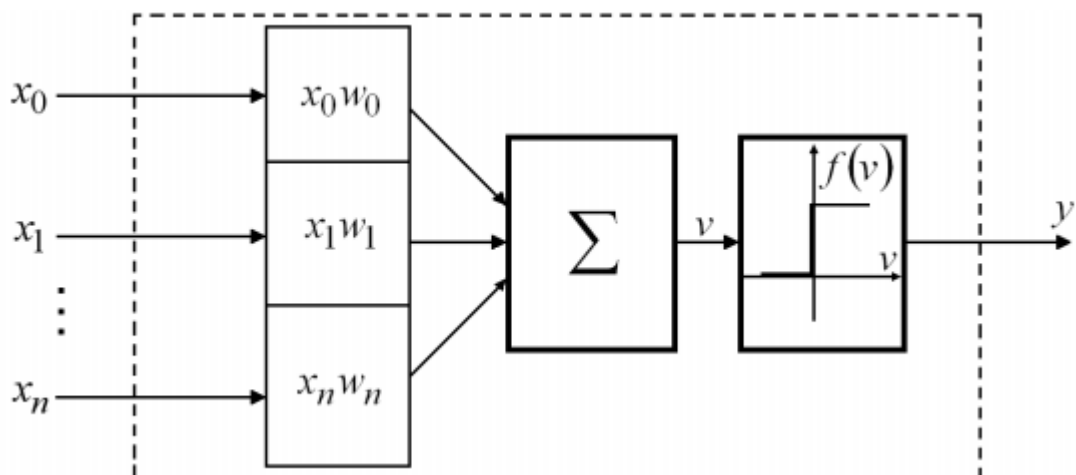


Рис. 1.2 Узагальнена структура штучного нейрона

Моделювання функцій нейронів відбувається шляхом масштабування вхідних сигналів  $(x_0, x_1, \dots, x_n)$  за допомогою вагових коефіцієнтів  $(w_0, w_1, \dots, w_n)$ . Одержані сигнали подаються на вхід суматора, який виконує їх обробку за формулою [11,12]:

$$v = \sum_{i=0}^n x_i w_i. \quad (1.1)$$

Вихідний сигнал суматора надходить на вхід блоку активації, в якому реалізується одна з можливих активаційних функцій. Змінна  $v$  є аргументом порогової активаційної функції. На виході блоку активації формується вихідний сигнал штучного нейрона  $y = f(v)$  [12].

Перша активаційна функція мала вигляд [12]:

$$y = \begin{cases} 1 & \text{при } v \leq 0 \\ 0 & \text{при } v > 0 \end{cases}.$$

У сучасній літературі зустрічається велика кількість парадигм штучних нейронних мереж, елементи яких реалізують різні активаційні функції. Найбільш поширеною є сигмоїдальна функція, що може бути представлена в дискретному та аналоговому варіанті [12].

Прикладом найпростішої активаційної функції є дискретний сигмоїд із заданим параметром активації  $a$ , що має назву функції Хевісайда або тетафункції [12]:

$$y = \begin{cases} 1 & \text{при } v \leq a \\ 0 & \text{при } v > a \end{cases}.$$

Якщо величина збудження  $v$  нейрона з активаційною функцією Хевісайда не перевищує значення параметра  $a$ , то нейрон залишається пасивним, а при перевищенні порога видає фіксоване значення функції, що приймається за логічну одиницю. Найпоширенішою нелінійною аналоговою активаційною функцією є сигмоїдальна логічна функція з параметрами,  $b$  і  $d$ , яку задають виразом [13]:

$$y = \frac{b}{c + e^{dv}}. \quad (1.2)$$

При одиничних значеннях параметрів  $b = 1$ ,  $c = 1$ ,  $d = -1$  одержуємо [13]:

$$y = \frac{1}{1 + e^{-v}}.$$

Амплітуда вихідного сигналу нейрона з даною активаційною функцією залежить від амплітуди вхідних сигналів. З рис. 1.3 видно, що крутизна S-подібної функції залишається високою тільки у деякому визначеному діапазоні [13].

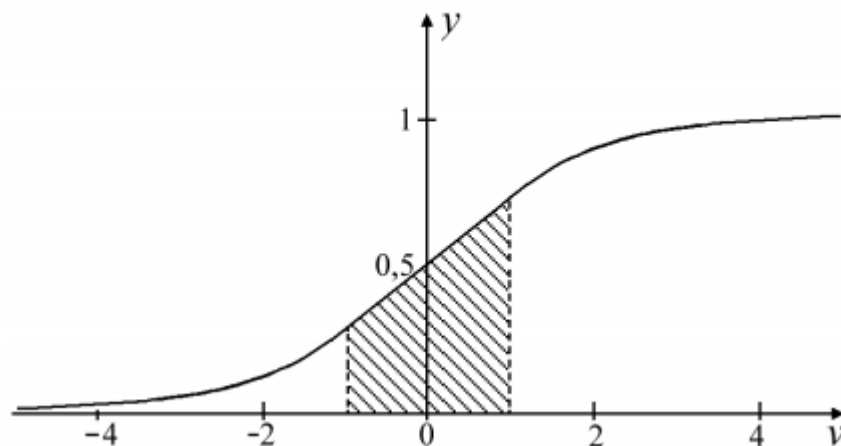


Рис. 1.3 Сигмоїдальна логічна функція

Штучна нейрона мережа – це набір штучних нейронів, з'єднаних між собою. Як правило, передатні функції всіх нейронів у мережі фіксовані, а ваги є параметрами мережі і можуть змінюватися. Деякі входи нейронів позначені як зовнішні входи мережі. Деякі виходи - як зовнішні виходи мережі. Подаючи будь-які числа на входи мережі, ми одержуємо якийсь набір чисел на виходах мережі. Таким чином, робота нейромережі складається в перетворенні вхідного вектора у вихідний вектор, причому це перетворення задається вагами мережі [14,15].

Хоча існують мережі, що містять лише один шар, або навіть один елемент, більшість реалізацій використовують мережі, що містять як мінімум три типи шарів - вхідний, схований і вихідний. Шар вхідних нейронів одержує дані або з вхідних файлів, або безпосередньо з електронних датчиків. Вихідний шар пересилає інформацію безпосередньо в зовнішнє середовище, до вторинного комп'ютерного процесу, або до іншого пристрою. Між цими двома шарами може бути кілька схованих шарів, що містять багато різноманітно зв'язаних нейронів. Входи і виходи кожного зі схованих нейронів з'єднані з іншими нейронами. Напрямок зв'язку від одного нейрона до іншого є важливим аспектом нейромереж. У більшості мереж кожен нейрон схованого шару одержує сигнали від усіх нейронів попереднього шару і звичайно від нейронів вхідного шару. Після виконання операцій над сигналами, нейрон передає свій вихід усім нейронам наступних шарів, забезпечуючи передачу сигналу вперед (feedforward) на вихід. При зворотному зв'язку, вихід нейронів шару направляється до нейронів попереднього шару. Сьогодні відома велика кількість нейронних структур та їх модифікацій, що орієнтовані на вирішення конкретного типу задач [14,15].

Найбільш відомі типи таких мереж показані на рис.1.4.

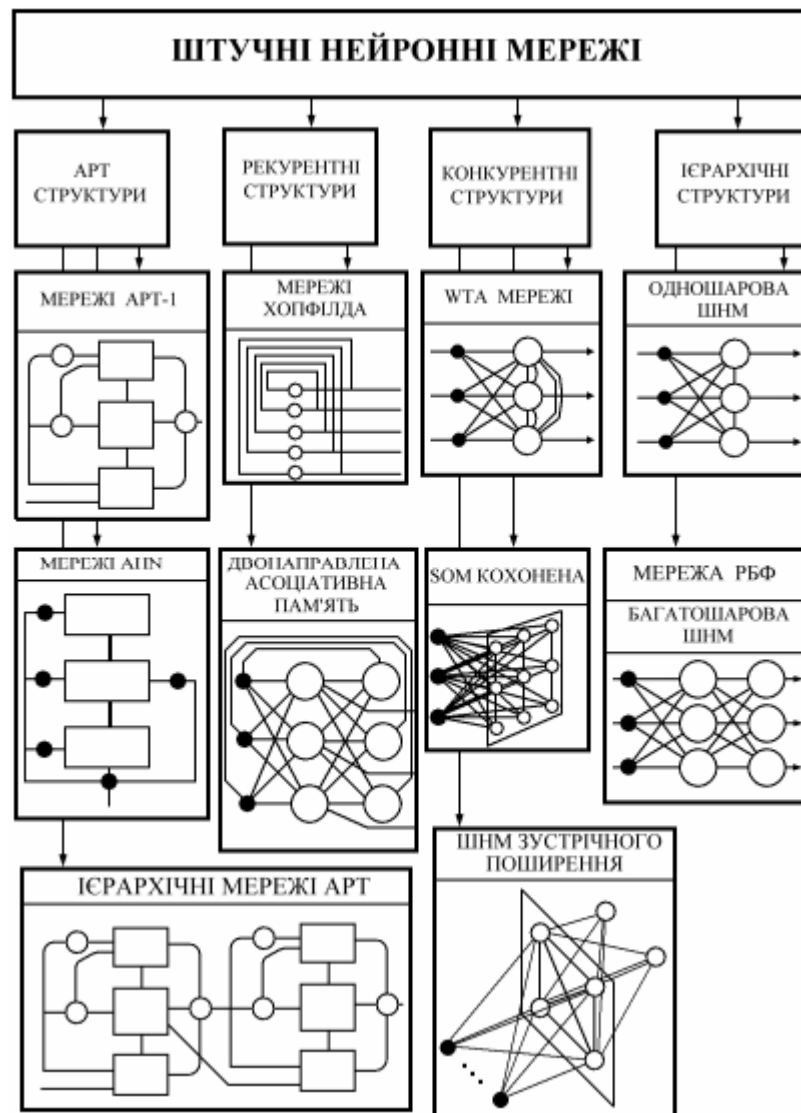


Рис. 1.4 Нейронні структури

ART – це структури які використовують теорію адаптивного резонансу.

WTA – це структури нейронного газу за конкурентним правилом Хебба.

РБФ – це штучні нейронні мережі, які використовують радіально базисні функції.

Мережі Хопхілда – це рекурентні структури зі зворотним зв'язком.

ШНМ зустрічного поширення – це структури в яких оновлення ваг проходить в обидва боки одночасно.

SOM структури – це структури з процедурами паралельного обчислення шарів, інші структури менш поширені та вузьконаправлені [14-16,20].

Розглянемо детальніше основні властивості ШНМ [14-16]:

- локальна обробка інформації в штучному нейроні, який є базовою структурною одиницею мережі;
- паралелізм, результатом якого є вирішення глобальної задачі шляхом представлення її у вигляді множини локальних задач, що тісно взаємодіють між собою;
- здатність до навчання, яке підвищує ефективність роботи мережі;
- здатність до розподіленого зберігання знань, які були одержані в ході навчання.

ШНМ задають у вигляді направлених графів, вершинами яких є нейрони, а ребрами позначені міжнейронні зв'язки. Архітектури сучасних нейронних мереж найчастіше поділяють на три категорії:

- мережі з повним набором міжнейронних зв'язків;
- мережі з фіксованим індексом оточення;
- мережі з пошаровою структурою.

У ШНМ із повним набором міжнейронних зв'язків забезпечується можливість взаємодії кожного нейрона мережі з будь-яким іншим. На рис. 1.5 наведений приклад повного з'єднання чотирьох нейронів [14-16].

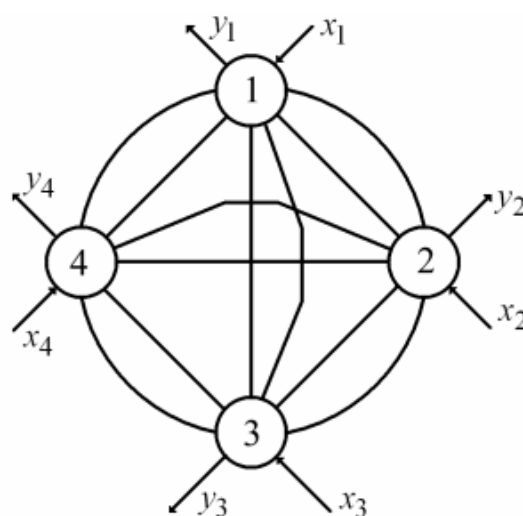


Рис. 1.5 Повне з'єднання нейронів у штучній нейронній мережі

Структури з повним з'єднанням є узагальненими структурами, оскільки всі інші довільні об'єднання нейронів можуть розглядатися як підмножини даної структури. Тому ШНМ із повним з'єднанням є універсальним середовищем для реалізації мережних алгоритмів. Широке використання таких структур обмежується недоліком, який полягає в значному зростанні кількості міжнейронних зв'язків при збільшенні кількості нейронів [14-16].

Задачі, які вирішують ШНМ, зводяться до апроксимації багатовимірних функцій, тобто побудови відображення  $F : x \rightarrow y$ . В залежності від вигляду активаційної функції формального нейрона дане відображення описує один з типів задач. У випадку застосування порогових активаційних функцій вихідні сигнали ШНМ мають дискретний характер, а задачі, що при цьому можуть бути розв'язні, називають задачами класифікації. При застосуванні формальних нейронів із сигмоїдальною або іншою неперервною активаційною функцією на ШНМ доцільно розв'язувати задачі регресії. Такий поділ є до деякої міри умовним, оскільки задачі класифікації можуть використовувати також неперервні функції активації, значення яких трактуються як імовірності приналежності до відповідних класів [14-16].

Отже, ШНМ виконують такі функції:

- класифікація,
- регресія,
- прогнозування часових послідовностей (рядів),
- кластерізація,
- асоціація,
- послідовність.

Перші три методи використовуються головним чином для прогнозування, у той час як останні зручні для опису існуючих закономірностей у даних.

Класифікація - найпоширеніша модель інтелектуального аналізу даних. З її допомогою виявляються ознаки, що характеризують групу, до якої належить



той або інший об'єкт. Це робиться за допомогою аналізу вже класифікованих об'єктів і формулювання деякого набору правил [14-16].

Регресійний аналіз використовується, коли відношення між змінними можуть бути виражені кількісно у виді деякої комбінації цих змінних. Отримана комбінація використовується для прогнозування чисельного значення, що може приймати цільова (залежна) змінна, що обчислюється на заданому наборі значень вхідних (незалежних) змінних. У найпростішому випадку, для цього використовуються стандартні статистичні методи, такі як лінійна регресія, але більшість реальних моделей ситуації не укладаються в її рамки [14-16].

Прогнозування часових послідовностей. Основою для будь-яких систем прогнозування служить історична інформація, збережена в інформаційних сховищах у виді часових рядів. Якщо можна побудувати математичну модель і знайти прототипи, що адекватно відображають цю динаміку, є ймовірність, що з їх допомогою можна прогнозувати поведінку системи в майбутньому. Прогнозування часових послідовностей дозволяє на основі аналізу поведінки часових рядів оцінювати майбутні значення прогнозованих змінних. Ці моделі повинні містити в собі особливі ознаки часу: ієрархія періодів (місяць-квартал-рік), особливі відрізки часу (п'яти- шести- або семиденний робочий тиждень), сезонність, свята й ін [14-16].

Кластеризація відрізняється від класифікації тим, що класи заздалегідь не задані і за допомогою моделі кластеризації засобу інтелектуальних обчислень самостійно створюють однорідні групи даних [14-16].

Асоціація сприяє здійснювати аналіз структур і застосовується, коли кілька подій зв'язані між собою .

Послідовність має місце, якщо існує ланцюжок зв'язаних у часі подій.

Найбільш поширеними є структуровані за шарами нейронні мережі, які, в залежності від свого функціонального призначення, можуть містити однотипні або різнотипні нейрони. Виходячи з шарової структури ШНМ, характер міжнейронних зв'язків має свої міжшарові та внутрішньшарові особливості. На рис. 1.6 показана класифікація видів міжнейронних зв'язків [14-16].



Рис. 1.6 Види міжнейронних зв'язків

Нейронні структури з повним з'єднанням можуть бути як одношаровими, так і багатошаровими. У одношаровій структурі з повним з'єднанням всі вхідні сигнали можуть поступати на всі нейрони. Класичною структурою даного типу є персептрон Розенблатта [14-16].

У випадку прямого міжшарового поширення інформація передається однонаправлено у напрямку зростання номера шару.

Пряме поширення в межах одного шару використовують у випадку, коли група нейронів даного шару з'єднана з попереднім шаром опосередковано через виділені нейрони.

Двонаправлене поширення допускає також зворотну передачу, що дозволяє створювати алгоритми, за якими враховувався б взаємний міжнейронний обмін.

Таким чином, однонаправленість зв'язків призводить до побудови виключно ієрархічних структур, у яких обробка інформації розподіляється по рівнях. За кожний рівень ієрархічної обробки інформації відповідає свій шар нейронів. Вихідна інформація більш високого рівня попереднього шару є

вхідною для нейронів наступного шару, який забезпечує глибший рівень обробки [14-16,21].

Двонаправлені міжшарові зв'язки необхідні для реалізації рекурентних структур, які дають можливість застосування ітераційних алгоритмів. Загальною рисою таких структур є те, що подальша передача інформації відбувається тільки у випадку завершення ітераційного процесу [14-16,21].

Двонаправлені зв'язки у межах одного шару використовуються для створення конкуруючих груп нейронів. При активації сигналом з попереднього шару кожен з нейронів передає сигнал активації нейронам своєї групи та сигнал гальмування всім іншим нейронам. В результаті конкурентоздатною стає та група нейронів, що одержала найбільше збудження [16].

### **1.3. Прогнозування генерації в MicroGrid на базі нейронних мереж**

Особливе значення серед функціонального ряду мають задачі прогнозування і прогнозування часових рядів, серед яких виділяються задачі з набором визначених специфічних ознак, тому варто провести їхню класифікацію. Задачі дослідження явищ, розвиток яких згодом зв'язано, можна поділити на кілька класів: По характеру основних ознак об'єкта [19,23]:

- прогнозування явищ, реалізації яких представлені у виді детермінованих часових рядів. Такі задачі, зокрема, можна розв'язувати шляхом застосування методів математичного аналізу;
- прогнозування явищ, реалізації яких представлені у виді індетермінованих часових рядів. Розв'язання цих задач традиційно здійснюється шляхом застосування методів теорії ймовірностей і математичної статистики.
- багатомірна задача; об'єкт або явище представлені декількома ознаками (графіки сейсмічної активності); задача прогнозування може бути розширена завдяки представленню даних у просторі (рис. 1.7) [19,23].

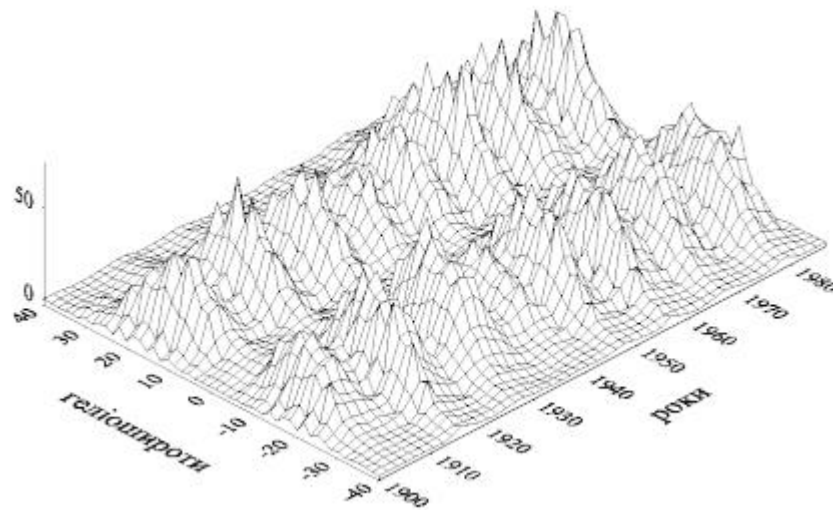


Рис. 1.7 Просторове відображення кількості сонячних плям у координатах геліоширота-час

З огляду на специфічний характер прогнозування часових рядів і визначений різнобій у термінології, будемо дотримуватись ряду означень. Передісторією ряду назвемо набір елементів часового ряду, що враховується для одного кроку прогнозування наступних елементів часового ряду. Однокрокове прогнозування зводиться до задач відображення у випадку, коли значення елементів передісторії можуть визначати лише один дискретний відлік вихідних величин. Багатокрокове прогнозування характеризується збільшенням дискретних відрахувань вихідної величини і, відповідно, збільшенням часу, на який здійснюється прогноз (час випередження  $T_{оп}$ ). При багатокроковому прогнозуванні  $T_{оп} = a \cdot R$ , де  $R$  - кількість кроків обчислення прогнозування;  $a$  - крок дискретизації вихідного параметра (наприклад, рік, місяць, день, і т.п.) [24-26].

Дані про поведінку об'єкта, ознаки якого зв'язані з часом, представлені як результати спостережень у рівномірні моменти часу. Для моментів часу  $t=1, 2, \dots, n$  дані спостережень здобувають вид часового ряду  $x(t_1), x(t_2), \dots, x(t_n)$ . Інформація про значення часового ряду до моменту  $n$  дозволяє давати оцінки параметрам  $x(t_{n+1}), x(t_{n+2}), \dots, x(t_{n+m})$ . Для прогнозування елементів часових рядів широко використовують так називаний метод "часових вікон" .

У залежності від кількості ознак, що представляють значення рядів при формуванні множин даних, виділимо задачі двох типів [25].

1. Однопараметрична задача прогнозування:

Нехай часовий ряд  $x(t)$  заданий моментами процесу  $x(t_1), x(t_2), \dots, x(t_i)$  у дискретні моменти часу  $t$ . Задамо ширину (кількість дискретних моментів) вхідного часового вікна  $m$ , ширину вихідного вікна  $p$ . Вхідне і вихідне вікна накладаються на дані ряду, починаючи з першого елемента (рис. 1.8) [25-26].

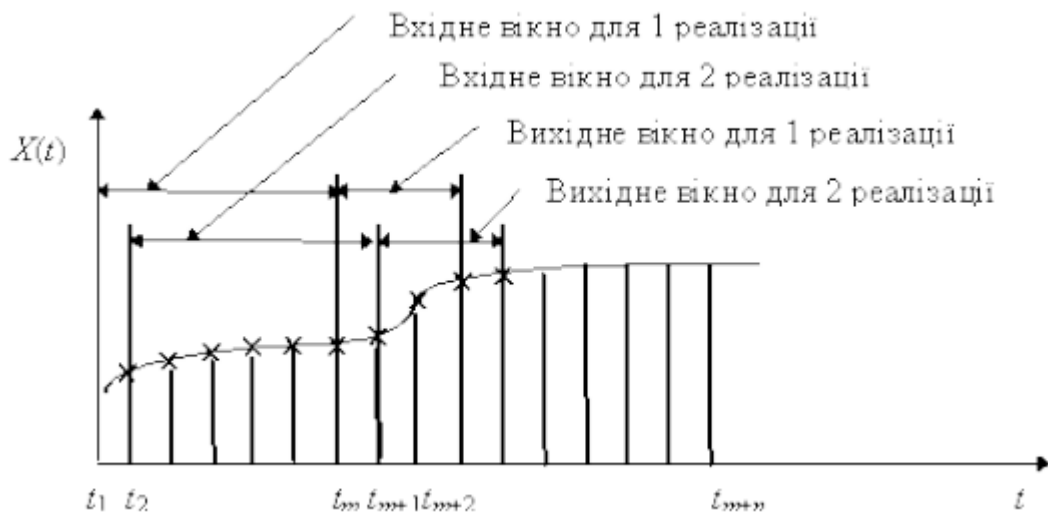


Рис. 1.8 Формування множин даних для однопараметричної задачі прогнозування по методу "часових вікон"

Вхідне вікно формує дані для входів нейронної мережі, а вихідне, відповідно, для виходів. Подібна пара вхідного і вихідного векторів приймається за одну реалізацію часового ряду. При зрушенні часових вікон по часовому ряду з кроком  $s$ , одержуємо другу і наступні реалізації. Значення ширини вікон і кроку зсуву повинні узгоджуватися з особливостями часового ряду, що забезпечується шляхом проведення експериментів. Нехай вхідне вікно має ширину  $m$ , вихідне вікно  $p = 1$ , крок зсуву  $s = 1$ . Тоді сформована множина значень для однопараметричної задачі буде мати вигляд, приведений в табл. 1.1. [25-26].

Таблиця 1.1

Множина даних для однопараметричної задачі

Входи				Вихід
$x(t_1)$	$x(t_2)$	...	$x(t_m)$	$x(t_{m+1})$
$x(t_2)$	$x(t_3)$	...	$x(t_{m+1})$	$x(t_{m+2})$
...	...	...	...	...
$x(t_i)$	$x(t_{i+1})$	...	$x(t_{i+m-1})$	$x(t_{i+m})$

## 2. Багатопараметрична задача прогнозування:

У багатомірних (багатопараметричних) задачах прогнозування підходи до рішення проблеми подібні (рис. 1.9) [25-26].

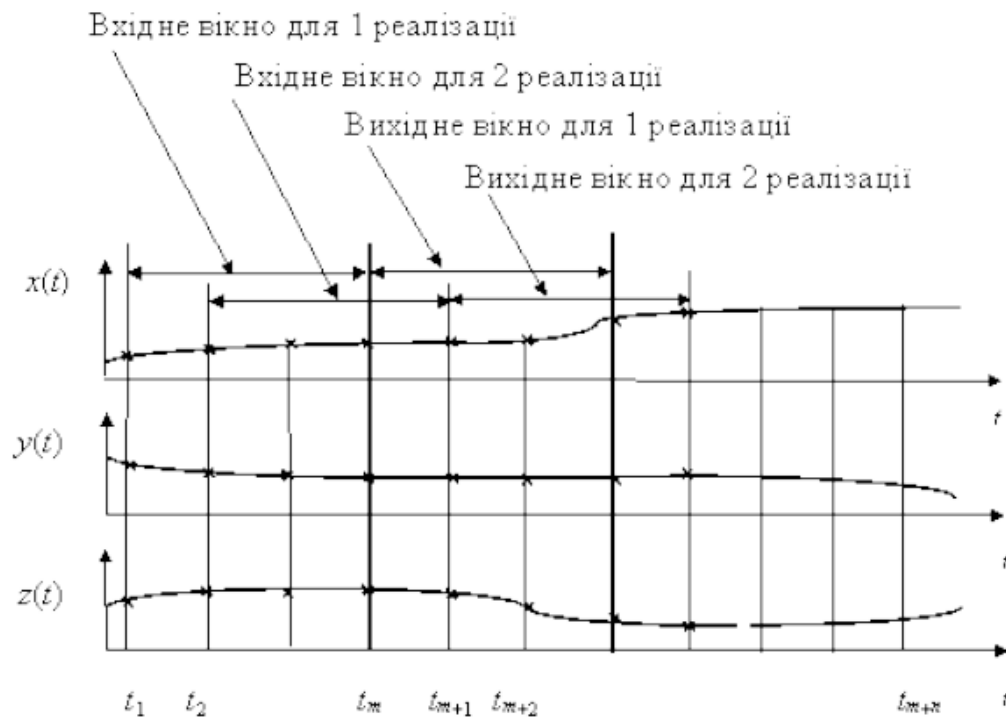


Рис. 1.9. Формування множин даних для багатопараметричної задачі

Нехай необхідно спрогнозувати взаємозалежні величини  $x(t)$ ,  $y(t)$ , ...,  $z(t)$ . Якщо прийняти ширину вхідного вікна  $m$ , вихідного  $p = 1$ , кроку зсуву  $s = 1$ , можна сформулювати множину даних, який показаний у табл. 1.2. [25 - 26]:

Таблиця 1.2

Множина даних для багатопараметричної задачі

Входи								
$x(t_1)$	$x(t_2)$	$x(t_m)$	$y(t_1)$	$y(t_2)$	$y(t_m)$	$z(t_1)$	$z(t_2)$	$z(t_m)$
$x(t_2)$	$x(t_3)$	$x(t_{m+1})$	$y(t_2)$	$y(t_3)$	$y(t_{m+1})$	$z(t_2)$	$z(t_3)$	$z(t_{m+1})$
...	...	...	...	...	...	...	...	...
$x(t_i)$	$x(t_{i+1})$	$x(t_{i+m-1})$	$y(t_i)$	$y(t_{i+1})$	$y(t_{i+m-1})$	$z(t_i)$	$z(t_{i+1})$	$z(t_{i+m-1})$
Виходи								
$x(t_{m+1})$	$y(t_{m+1})$	$z(t_{m+1})$						
$x(t_{m+2})$	$y(t_{m+2})$	$z(t_{m+2})$						
$x(t_{m+3})$	$y(t_{m+3})$	$z(t_{m+3})$						
$x(t_{i+m-1})$	$y(t_{i+m-1})$	$z(t_{i+m-1})$						

Функціонування неймережі здійснюється відповідно до показаного методу часових вікон, зберігаючи значення ширини вікон і кроку зсуву. Конкретизація підходів до реалізації прогнозування значною мірою залежить від особливостей досліджуваного явища [25-26].

Задача однокрокового прогнозування зводиться до задачі відображення, коли один вхідний вектор відображається у вихідний (рис. 1.10).

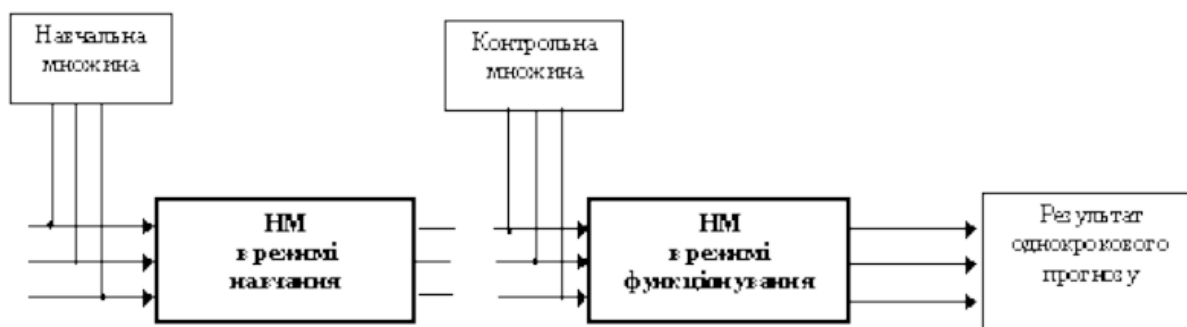


Рис. 1.10 Послідовність використання неймережі для задач передбачення

У випадку однопараметричної задачі передбачення навчальна множина до моменту  $n$ , за умови  $m=3$ ,  $p=1$ ,  $s=1$ , буде мати вигляд приведений у табл. 1.3 [26].

Таблиця 1.3

Множина даних для однопараметричної задачі

Входи			Вихід
$x(t_1)$	$x(t_2)$	$x(t_3)$	$x(t_4)$
$x(t_2)$	$x(t_3)$	$x(t_4)$	$x(t_5)$
...	...	...	...
$x(t_{n-3})$	$x(t_{n-2})$	$x(t_{n-1})$	$x(t_n)$

У режимі навчання встановлюються коефіцієнти ваг зв'язків, після чого стає можливим перехід до режиму функціонування. Для передбачення на входи нейромережі надходять значення останньої реалізації навчальної множини  $x(t_{n-2})$ ,  $x(t_{n-1})$ ,  $x(t_n)$ . На виході формується прогнозована величина  $x^*(t_{n+1})$ .

Для багатопараметричної задачі передбачення на входи навченої нейромережі подаються вектори  $x(t_{n-2})$ ,  $y(t_{n-2})$ ,  $z(t_{n-2})$ ,  $x(t_{n-1})$ ,  $y(t_{n-1})$ ,  $z(t_{n-1})$ ,  $x(t_n)$ ,  $y(t_n)$ ,  $z(t_n)$ . На виходи нейромережі надходять передбачені величини  $x^*(t_{n+1})$ ,  $y^*(t_{n+1})$ ,  $z^*(t_{n+1})$ , що відкладаються у вихідний вектор передбачених даних.

Показаний режим є однокроковим, що працює в режимі відображення (реальний вхід - прогнозований вихід). Передбачення застосовують для моделювання дискретних послідовностей, що не зв'язані часом. З огляду на специфіку часових рядів, такий тип прогнозу не завжди доцільний, але у визначених випадках короткострокових прогнозів їм можна скористатися [26].

### Багатокрокове прогнозування

Багатокрокове прогнозування застосовують лише для явищ, ознаки яких представлені у виді часових рядів. Для однопараметричної задачі прогнозування



навчальна множина буде мати вигляд приведений у табл. 1.3. Під час навчання мережа налаштовує коефіцієнти ваг зв'язків і поліномів передатних функцій, що надалі і визначає режим функціонування. Багатокрокове прогнозування часового ряду здійснюється в такий спосіб (рис. 11). На входи нейромережі подається вектор відомих значень  $x(tn-2)$ ,  $x(tn-1)$ ,  $x(tn)$ . На виході формується прогнозована величина  $x^*(tn+1)$ , що визначає вектор прогнозованих виходів і одночасно додається до значень навчальної множини, тобто, приймається як достовірна. Далі на входи подається вектор  $x(tn-1)$ ,  $x(tn)$ ,  $x^*(tn+1)$ , а на виході виходить  $x^*(tn+2)$  і наступні прогнозовані значення [26].

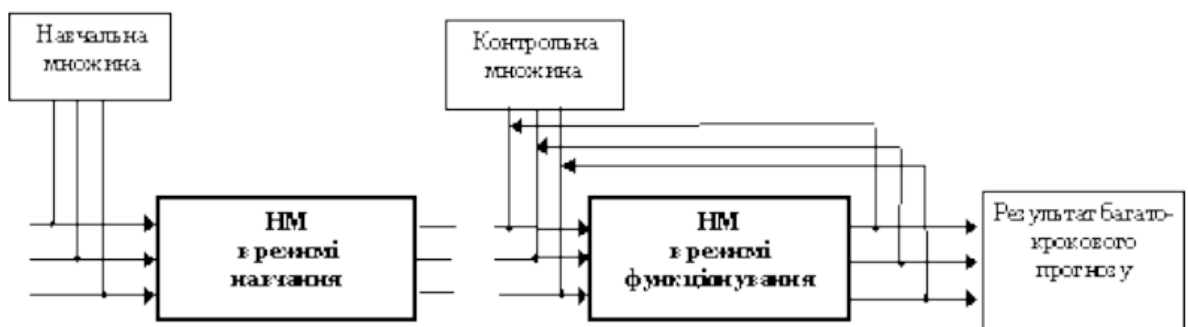


Рис. 1.11 Послідовність використання нейромереж для задач багатокрокового прогнозування

Для багатомірної задачі прогнозування на входи навченої нейромережі подаються вектори  $x(tn-2)$ ,  $y(tn-2)$ ,  $z(tn-2)$ ,  $x(tn-1)$ ,  $y(tn-1)$ ,  $z(tn-1)$ ,  $x(tn)$ ,  $y(tn)$ ,  $z(tn)$ . На виході продукуються величини  $x^*(tn+1)$ ,  $y^*(tn+1)$ ,  $z^*(tn+1)$ , що формують вектор вихідних значень і послідовно додаються до значень навчальної множини. При зрушенні вікна на крок, вихідні дані, які спродуковані мережею, сприймаються як реальні і беруть участь у прогнозуванні наступного значення виходу, тобто на входи подаємо вектор  $x(tn-1)$ ,  $y(tn-1)$ ,  $z(tn-1)$ ,  $x(tn)$ ,  $y(tn)$ ,  $z(tn)$ ,  $x^*(tn+1)$ ,  $y^*(tn+1)$ ,  $z^*(tn+1)$ , а на виході одержуємо  $x^*(tn+2)$ ,  $y^*(tn+2)$ ,  $z^*(tn+2)$  і наступні прогнозовані значення [26].

Багатокрокове прогнозування дозволяє робити коротко- і середньострокові прогнози, оскільки істотний вплив на точність має нагромадження похибки на кожному кроці прогнозування. При застосуванні довгострокового

багатокрокового прогнозування спостерігається характерне для багатьох прогнозуючих систем поступове загасання процесу, фазові зрушення й інші перекручування картини прогнозу. Такий тип прогнозування підходить для стаціонарних часових рядів з невеликою випадковою складовою [26].

Багатокрокове прогнозування з перенавчанням нейромережі на кожному кроці прогнозу

Швидкі неітераційні алгоритми навчання дозволяють запропонувати новий тип багатокрокового прогнозу, що може бути застосований при довгострокових прогнозах зі збереженням задовільної точності прогнозування.

Аналогічно з попереднім алгоритмом прогнозування на входи мережі в режимі функціонування надходить остання реалізація навчальної множини  $x(tn-2), x(tn-1), x(tn)$ . Прогнозоване значення виходу  $x^*(tn+1)$  відкладається у векторі прогнозованих вихідних значень і в якості достовірного додається до реальних значень навчальної множини. Навчальна множина збільшується на одне часове вікно. Відбувається процес перенавчання мережі на збільшеній навчальній множині, під час якого визначаються нові вагові коефіцієнти  $k$  синаптичних зв'язків і поліномів передатних функцій нейронів (рис. 1.12) [25-26].

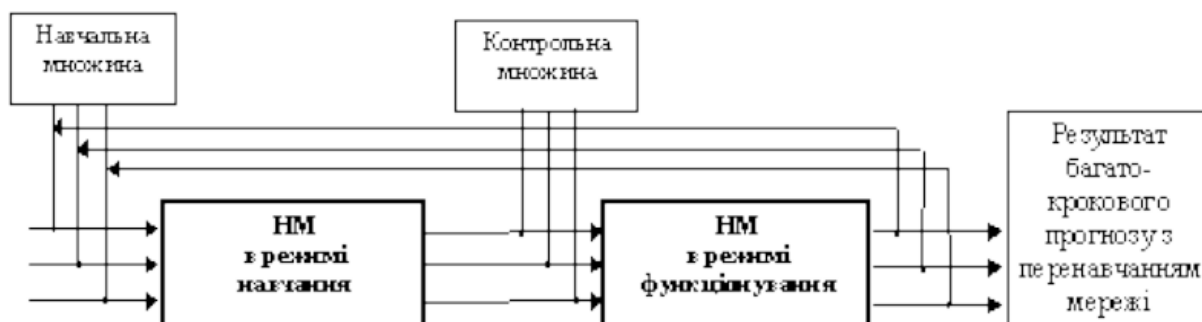


Рис. 1.12 Послідовність використання нейромережі для задач багатокрокового прогнозування з перенавчанням

Реалізація  $x(tn-1), x(tn), x^*(tn+1)$ , як значення наступного вхідного вікна подається на входи мережі в режимі функціонування. Мережа продукує нове вихідне значення  $x^*(tn+2)$ , що також відкладається у вектор спродукованих виходів і додається до реальних значень навчальної множини, з метою

подальшого перенавчання мережі і установлення відновлених коефіцієнтів поліномів передатних функцій і синаптичних зв'язків. Ітераційна процедура перенавчання поширюється до прогнозованого значення  $x^*(t)$ . Такий підхід дозволяє при великих інтервалах випередження усунути загасання прогностичних властивостей мережі за рахунок постійного коректування вагових коефіцієнтів синаптичних зв'язків. Відзначимо, що алгоритм багатокрокового прогнозування з перенавчанням мережі для традиційних мереж прямого поширення з ітераційним навчанням є практично нездійсненним через великі часові затримки, необхідні на перелагоджування коефіцієнтів мережі.

#### 1.4. Основи моделей нейронних мереж

Одною з початкових штучних нейронних мереж, що була згідно категорії віднесена до одношароврі, було персептрон Розенблатта (рис. 1.13) [27].

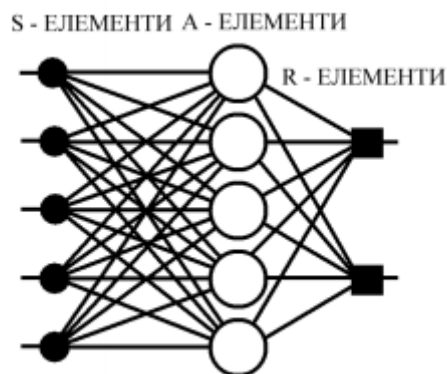


Рис. 1.13 Персептрон Розенблатта

Так як персептрон описувався Розенблаттом як структура що нагадує структуру мозку, то нейрони в цій структурі ототожнювалися з нейронами звичайної рефлекторної мережі. S-елементи виконують роботу сенсору та потребують інформації про навколишнє середовище та направляють її до А-елементів, які були формальними нейронами з звичайною пороговою активаційною функцією. Щоб сформувати реакцію персептрона були необхідні R-елементи, що мали конкретні початкові ваги щоб отримати інформацію про

вагу окремого А-елемента. Сучасною моделлю персептрона Розенблатта є одношарова нейронна мережа прямого поширення, яку показано на рис. 1.14 [27].

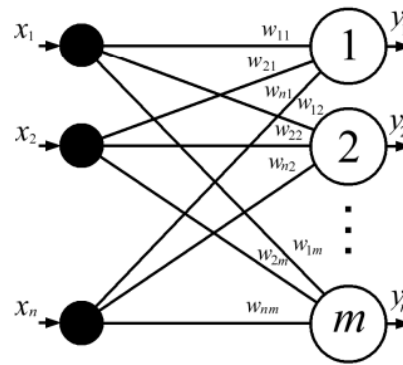


Рис. 1.14 Одношарова нейронна мережа

Ця мережа має в складі  $m$  нейронів, так що всі разом приймають вхідний вектор сигналів  $X = (x_1, \dots, x_i, \dots, x_n)$ . Щоб провести дуплікацію  $x_i$  цей вектор використовується особливий пристрій, що представлені лівіше від нейронів. Цей пристрій не виконує аналіз даних, так як не є власне шаром нейронної мережі. У зв'язку з структурою моделі звичайного нейрона, кожен з їх вхідних даних мультиплікують з ваговим коефіцієнтом  $w_{ij}$ , і де  $i$  — це справжній номер частини вектора  $X$ , а  $j$  — справжній номер нейрона. Вся кількість вагових коефіцієнтів одношарової нейронної мережі утворює матрицю вагових коефіцієнтів [27-28]:

$$W = \begin{pmatrix} w_{11} & \dots & w_{1j} & \dots & w_{1m} \\ \vdots & & \vdots & & \vdots \\ w_{i1} & \dots & w_{ij} & \dots & w_{im} \\ \vdots & & \vdots & & \vdots \\ w_{n1} & \dots & w_{nj} & \dots & w_{nm} \end{pmatrix}.$$

Так, вектор аргументів обчислюється як добуток  $V = XW$ , а вектор вихідних даних є вектором елементів активаційних функцій:

$$Y = F(V) = \begin{cases} f_1(v_1) \\ \dots \\ f_j(v_j) \\ \dots \\ f_m(v_m) \end{cases}.$$

Ці та інші функції дуже схожі для усіх нейронів та утворюють мережу. В такій ситуації її прийнято вважати гомогенною. Мережа, де вигляд активаційної функції залежить від номерів нейронів, вважають гетерогенною. На рис. 7 показано просту одношарову нейронну мережу, в ній кожен нейрон бере участь в аналізі більшості частин вхідних векторів. Цей прийом не часто є економічно та технічно виправданим. Саме тому при вирішенні конкретної задачі може використовуватись архітектура за структурою елементів, де є певна підмножина з повною елементозв'язністю [27-28].

Реальні приклади застосування одношарових нейронних мереж під вирішення справжніх задач зіштовхнулись з певними перепонами, зв'язані з проблематикою лінійної роздільності. Простим подоланням труднощів було використання багатошарових ШНМ, які наслідують багатошарові моделі мозку. Опишемо ієрархічну модель, де нейрони упорядковані пошарово. Така структура містить в собі  $m$  нейронів початкового (прихованого) шару, та разом з тим можуть мати початковий вектор ваг  $X = (x_1, \dots, x_i, \dots, x_n)$ , і  $k$  нейронів іншого (вихідного) шару. Ця модель представлена на рис. 1.15 і є загальною моделлю багатошарової нейронної мережі прямого поширення.

В нейронних мережах прямого поширення синаптичні зв'язки побудовані так, що кожен нейрон цього рівня ієрархії отримує данні особисто від конкретної вагової множини нейронів, що знаходяться нижче від данного рівня. Ім'я мережі говорить, що там знаходиться окремий напрям передачі інформації, та рух відбувається, від початку і далі, проходячи крізь один або більше прихованих шарів до останнього (виходу) шару [27-28].

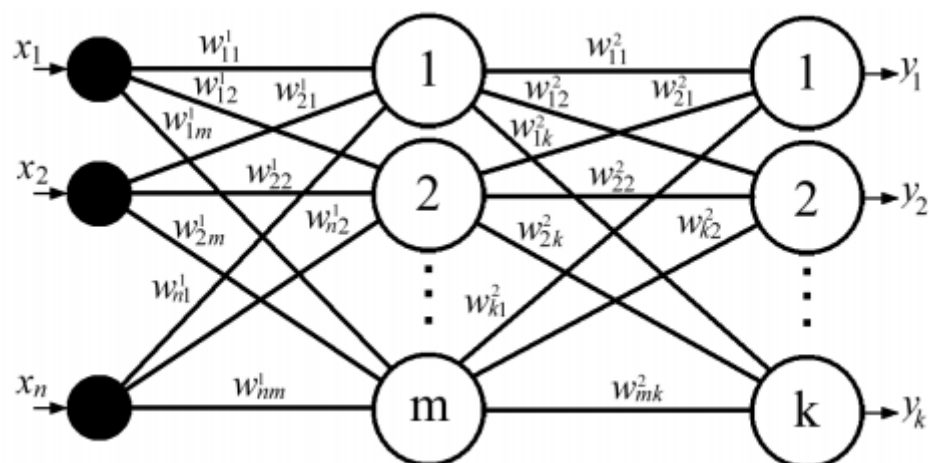


Рис. 1.15 Багатошарова нейронна мережа прямого поширення.

Очевидно, що багатошарова нейронна мережа є продуктом каскадного поєднання одношарових мереж з матрицями вагових коефіцієнтів  $W^1, W^2, \dots, W^p$ , де  $p$  — кількість шарів нейронної мережі.

Для ситуації лінійності активаційних функцій багатошарова нейронна мережа може бути зведена до подібної одношарової з матрицею вагових коефіцієнтів  $W = W^1 * W^2 * \dots * W^p$ , тому реалізація таких моделей є доцільною лише при використанні нелінійних активаційних функцій в нейроні.

Моделі мереж прямого поширення, у яких відсутні зворотні зв'язки, а це інформація яка проходить шар за шаром лише прямо, створюючи основну вагу в більшості нейронної мережі.

Також важливим є результат використання конкурентних моделей. В структурах цього типу затухаючий ітераційний процес відбувається в межах одного шару нейронів, а ітераційна формула завжди має властивість падіння рівня вихідного сигналу, та сприяє затухання низьких вихідних даних до показника, меншого за поріг чутливості.

Взагалі іншою є ситуація, коли структура нейронної мережі допускає зворотні зв'язки, тобто, коли обчислення в нейроні даного шару відбуваються з урахуванням попереднього стану цього ж шару. Мережі такого типу називають рекурентними та говорять про динамічний характер їх функціонування. Динаміка зміни станів сукупності взаємопов'язаних об'єктів традиційно вимагає

визначення правила взаємної координації дій цих об'єктів у часі та просторі. Коли встановлюється послідовність спрацьовування нейронів одного шару, то таку нейронну мережу називають синхронною. У випадку, коли час спрацьовування кожного з нейронів не регламентується, мережу називають асинхронною. Синхронні мережі можуть бути паралельними, послідовними та паралельно-послідовними. Для паралельних мереж визначальною є властивість одночасного спрацьовування всіх нейронів одного шару, для послідовних мереж задають послідовність спрацьовування нейронів одного шару, а для паралельно-послідовних мереж визначається послідовність спрацьовування кластерів нейронів. Про асинхронні мережі говорять, що вони є паралельними, маючи на увазі те, що такі мережі допускають одночасне функціонування всіх нейронів.

Структура рекурентної нейронної мережі, яка показана на рис. 1.16, вперше була запропонована в [27-28].

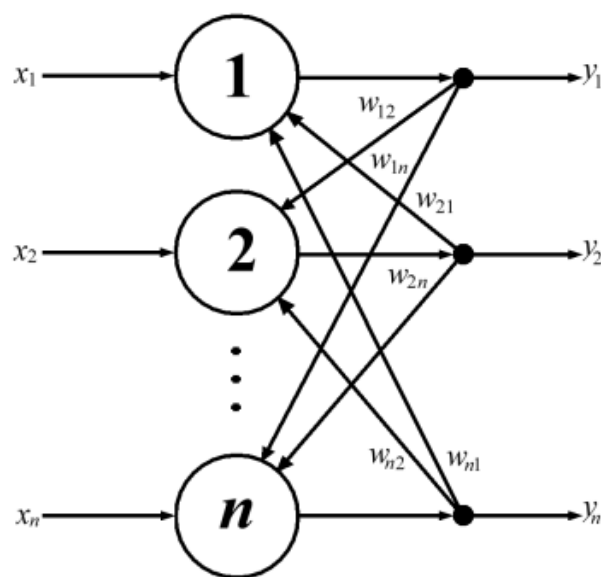


Рис. 1.16 Нейронна мережа Хопфілда

Вона складається з одного шару нейронів, виходи яких через спеціальні пристрої з'єднані зі входами всіх нейронів цього ж шару, крім тих зв'язків, що з'єднують вихід нейрона з його власним входом. Формування аргументу активаційної функції  $v_j$  довільного нейрона  $j$  відбувається за формулою [27-28]:

$$v_j = \sum_{\substack{i=0 \\ i \neq j}}^n w_{ij} y_i + x_j, \quad (1.3)$$

Бінарна модель базується на використанні порогової активаційної функції, яку задають виразом [27-28]:

$$y_i = f(v_j) = \begin{cases} 1 & \text{при } v_j > T_j \\ 0 & \text{при } v_j < T_j \\ y_j & \text{при } v_j = T_j \end{cases},$$

де,  $T_j$  позначається величина порога чутливості довільного нейрона  $j$  мережі Хопфілда. Такий вигляд активаційної функції був запропонований Хопфілдом. У сучасній літературі частіше зустрічається порогова активаційна функція зі зміною знака:

$$y_i = f(v_j) = \begin{cases} 1 & \text{при } v_j > T_j \\ -1 & \text{при } v_j < T_j \\ y_j & \text{при } v_j = T_j \end{cases}.$$

Якщо для виходу  $j$  у кожного нейрона поставити у відповідність двійковий розряд, то поточний стан мережі Хопфілда може бути виражений числом у двійковій системі числення, а множина допустимих сусідніх станів є множиною чисел з одиничною відстанню Хеммінга. Якщо стани нейронної мережі позначити точками з відповідними координатами у просторі та з'єднати дугами ті точки, між якими допустимий перехід, одержимо граф переходів мережі. Для мережі, що складається з трьох нейронів, такий граф має вигляд куба (рис. 1.17). Мережі з кількістю нейронів більше трьох утворюють гіперкуби вищих порядків, що не мають графічної інтерпретації. У загальному випадку множина сусідніх вершин  $N$  для довільної вершини  $a$  визначається з виразу [27-28]:

$$N(a) = \{a \mathbf{XOR} 2^{i-1}\}_{i=1}^n, \text{ де } i = 0, 1, \dots, 2^n - 1. \quad (1.4)$$



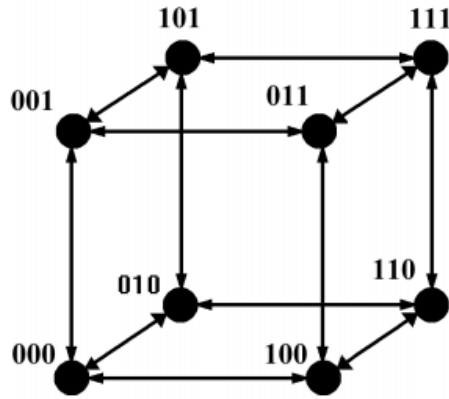


Рис. 1.17 Граф переходів мережі з 3 нейронами

Функціонування мережі Хопфілда полягає в пересуванні вздовж ребер переходів, поки мережа не досягне стійкого стану, умови наявності якого вперше сформульовані в [18]. Мережа Хопфілда має стійкий стан у випадку, коли матриця вагових коефіцієнтів  $W$  симетрична і має нулі на головній діагоналі, тобто  $w_{ij} = w_{ji}, w_{ii} = 0$ .

Для пояснення факту стійкості мережі за згаданих умов введемо функцію енергії довільної пари нейронів:  $e_{ij} = -w_{ij}y_i y_j - x_j y_j + T_j y_j$ .

Виходячи з умови симетричності  $w_{ij} = w_{ji}, w_{ii} = 0$  матриці вагових коефіцієнтів, загальну енергію  $E$  нейронної мережі визначають за функцією Ляпунова [27-28]:

$$E = -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n w_{ij} y_i y_j - \sum_{j=1}^n x_j y_j + \sum_{j=1}^n T_j y_j. \quad (1.5)$$

Загальна енергія мережі Хопфілда за умов  $w_{ij} = w_{ji}, w_{ii} = 0$ , повинна завжди або зменшуватися, або залишатися незмінною. Стійкий стан мережі відповідає мінімуму її енергії.

Поверхня енергії в просторі станів мережі має складну форму, яка характеризується великою кількістю локальних мінімумів. У випадку застосування мережі Хопфілда в ролі запам'ятовуючого середовища стійкі стани, що відповідають локальним мінімумам енергії, інтерпретують як образи пам'яті. Під час переходу мережі від одного стану до такого, що

характеризується меншим рівнем енергії, завжди відбувається пошук сусіднього стану за формулою (4). Тому мережа завжди знаходить локальний енергетичний мінімум, який розміщується на мінімальній відстані Хеммінга від початкового стану [27-28,18]. Нехай вектор вхідних сигналів  $X$  відповідає ідеальному образу пам'яті, тобто локальному мінімуму енергії мережі. Тоді надходження на вхід мережі вектора  $X_i$  приводить до еволюції мережі у напрямку локального мінімуму, що відповідає відновленню ідеального образу по його неповній копії. Така пам'ять відрізняється від тієї, що використовується в традиційних обчислювальних структурах. Вона більше нагадує людську пам'ять, для якої характерний асоціативний, а не адресний принцип запам'ятовування інформації. Асоціативна пам'ять є значно стійкішою щодо різного роду помилок та шумів за рахунок того, що має здатність відновлення ушкодженого образу, на відміну від адресної пам'яті, у якій помилка адресації призводить до неправильного зчитування значних фрагментів. Недоліком асоціативної пам'яті, яка реалізується на мережі Хопфілда, є незначний її об'єм, пропорційний кількості нейронів. У цьому випадку об'єм адресної пам'яті пропорційний  $2^n$ , де  $n$  — кількість двійкових розрядів.

Згаданий недолік та ряд інших, які характерні для побудованої на основі класичної мережі Хопфілда асоціативної пам'яті, роблять її малопридатною для практичного використання [27-28,18].

### **Висновки до розділу 1**

1. MicroGrid — комплексна електротехнічна система, що складається з різних підсистем, які поєднані та взаємодіють між собою.
2. Штучна нейрона мережа – це набір штучних нейронів, з'єднаних між собою в деяку структуру і виконують певну задачу.
3. Проведений аналіз задачі прогнозування та використано багатокрокове з перенавчанням.
4. Наведено основні моделі нейронних мереж персептрон Розенблата, одношарову та багатошарову прямого поширення, а також ШНМ Хопфілда.

## РОЗДІЛ 2. МОДУЛІ ПРОГНОЗУВАННЯ НА БАЗІ НЕЙРОННИХ МЕРЕЖ

### 2.1. Рекурентна нейронна мережа CRBM

Раніше розглядалась класична рекурентна структура, поведінку нейронів у якій визначають певні детерміновані правила. Виходячи з цих правил, можна визначити стан мережі в кожний фіксований момент часу. Еволюція стійких детермінованих рекурентних мереж завжди відбувається у напрямку локального мінімуму енергії. Ця властивість є корисною у випадку, коли нейромережа моделює функції асоціативної пам'яті. Коли ж ставиться задача пошуку глобального мінімуму, стійкий стан мережі у локальному мінімумі є небажаним явищем. Для розв'язування подібних задач використовують стохастичні рекурентні мережі. Принциповою відмінністю стохастичних алгоритмів є відмова від заборони переходу мережі у стан з більш високим рівнем енергії. Модель, що описує роботу стохастичної мережі, нагадує процес застигання металу, кожен атом якого знаходиться в неупорядкованому русі і має різний запас енергії, але загальна температура прямує до нуля. Імовірність переходу між енергетичними рівнями визначається за формулою Больцмана [30]:

$$P(E) = \frac{1}{1 + \exp\left\{-\frac{E_2 - E_1}{T}\right\}}, \quad (2.6)$$

де  $P(E)$  — імовірність переходу системи зі стану з енергією  $E_1$  у стан з енергією  $E_2$ ;  $T$  — температура за шкалою Кельвіна.

При великих значеннях температури величина імовірності  $P(E) = \frac{1}{2}$ , тобто наступний стан системи, не залежить від її попереднього стану, що відповідає хаотичній поведінці. Зі зменшенням температури величина імовірності прямує до одиниці, і поведінка мережі вже в такому стані збігається з поведінкою детермінованої мережі. Такі мережі показали високу ефективність для розв'язування задач пошуку глобального мінімуму функції та передбачення. Прикладом такої мережі є Обмежена машина Больцмана (ОМБ, англ. CRBM – Conditional Restricted Boltzmann Machine) [30].

ОМБ являє собою стохастичну штучну нейронну мережу, навчання якої здійснюється «без вчителя» за алгоритмом зворотного розповсюдження помилки. Як і інші НМ, ОМБ складається з видимих і прихованих шарів нейронів.

Мережа «Машина Больцмана» (Boltzmann mashine) схожа по функції і дії на мережу Хопфілда і включає поняття "модельованого віджигу" для пошуку в просторі станів шару образів глобального мінімуму [30,31].

Экли (Ackley), Хинтон (Hinton) і Сейновски (Sejnowski) розробили правило больцмановського навчання в 1985 р. Подібно мережі Хопфілда, машина Больцмана має простір станів, що базується на вагах з'єднань у шарі образів. Процеси навчання мережі, наповненої образами, включає згладжування рельєфу простору станів. Машина Больцмана моделює віджиг металу, що додається до процесу навчання мережі. Як і при фізичному віджигу, температура починається з великих значень і зменшується з часом. Збільшена температура додає збільшений шумовий коефіцієнт до будь-якого нейрона в шарі образів. Переважно, кінцевою температурою буде нуль. Для досягнення оптимального рішення доцільно на нижчих температурах додавати більше ітерацій. Машина Больцмана, навчаючись на високій температурі, поводить як випадкова модель, а на низьких температурах вона як детермінована модель. Через випадковий компонент в віджиговому навчанні, нейрон може прийняти нове значення стану, що збільшується швидше, ніж зменшується загальний простір станів. Імітація фізичного віджигу дозволяє просуватися до глобального мінімуму, уникаючи локальний. Як і в мережі Хопфілда, мережі може бути представлений частковий образ для відновлення відсутньої інформації [30].

Алгоритм функціонування мережі

1. Визначити змінну  $T$ , що представляє штучну температуру.
2. Пред'явити мережі множину входів і обчислити виходи і цільову функцію.
3. Дати випадкову зміну вагам і перерахувати вихід мережі і зміну цільової функції відповідно до зміни ваг.

4. Якщо цільова функція зменшилася, тоді нова множина ваг зберігається. Якщо зміна ваг приводить до збільшення цільової функції, то ймовірність збереження цієї зміни обчислюється за допомогою розподілу Больцмана:

$$P(c) = e^{\frac{-c}{kT}}.$$

де  $P(c)$  - ймовірність зміни  $c$  у цільовій функції;  $k$  - константа, аналогічна константі Больцмана, вибирається в залежності від задачі;  $T$  - штучна температура.

Метод стохастичного машинного навчання тут застосовується для моделювання часових рядів, які характеризуються значним ступенем нестаціонарності та фактично не можуть бути апроксимовані аналітичними залежностями. В процесі навчання штучної нейронної мережі змінюються вагові коефіцієнти нейронів різних шарів. Відповідно змінюється і зважена сума значень вихідних сигналів нейронів, яка надалі надходить на блок активації. В залежності від функції, закладеної у блок активації, формується вихідний сигнал. Отже, результатом кожної фази навчання є зміна вагових коефіцієнтів та параметрів функції активації.

Зв'язки між нейронами різних шарів забезпечують проходження сигналу від входу до виходу. У з'єднанні виходу  $i$ -го нейрону із входом  $j$ -го перший вважається попередником другого, а другий - наступником першого. З'єднанню цих двох нейронів призначається вага  $w_{ij}$ .

Вагова функція ОМБ має наступний виглядх [29]:

$$E(v, h) = - \sum_{i=1}^m \sum_{j=1}^n v_i h_j w_{ij} - \sum_{i=1}^m v_i a_i - \sum_{j=1}^n h_j b_j ,$$

де  $i$  – номер нейрону видимого шару,  $j$  – номер нейрону прихованого шару,  $w_{ij}$  – вага з'єднання;  $v_i$  і  $h_j$  - стан  $i$ -го видимого і  $j$ -го прихованого нейрону;  $a_i$  і  $b_j$  – вагові коефіцієнти упереджень видимих і прихованих шарів.

Вираз  $\sum_{i=1}^m \sum_{j=1}^n v_i h_j w_{ij}$  визначає загальну енергію зв'язку між нейронами різних шарів,  $\sum_{i=1}^m v_i a_i$  – енергію видимого шару,  $\sum_{j=1}^n h_j b_j$  – енергію прихованого шару.

Спільна ймовірність стану прихованого і видимого шарів визначається виразом [29]:

$$P(v, h) = \frac{\exp(-E(v, h))}{Z} = \frac{1}{Z} \exp\left(-\sum_{i=1}^m \sum_{j=1}^n v_i h_j w_{ij} - \sum_{i=1}^m v_i a_i - \sum_{j=1}^n h_j b_j\right),$$

де  $Z = \sum_{v, h} \exp(-E(v, h))$  – це сума ваг по всім можливим парам видимих и прихованих з'єднань.

Ймовірність стану видимого шару визначається шляхом додавання ймовірностей станів прихованого шару [29]:

$$p(v) = \sum_h P(v; h).$$

Після перетворень отримаємо:

$$p(v) = \frac{1}{Z} \sum_h \left( \exp\left(-\sum_{i=1}^m \sum_{j=1}^n v_i h_j w_{ij} - \sum_{i=1}^m v_i a_i - \sum_{j=1}^n h_j b_j\right) \right).$$

Для будь-якого прихованого нейрона  $j$  умовна ймовірність стану визначається як [29]:

$$p(h_{j|v}) = \text{sig}(b_j + \sum_i v_i w_{ij}).$$

Для будь-якого видимого нейрона  $i$  умовна ймовірність стану визначається як:

$$p(v_{i|h}) = \text{sig}(a_i + \sum_j h_j w_{ij}).$$

де  $\text{sig}(*)$  – сигмоподібна функція активації, що визначається виразом:

$$\sigma(u) = \frac{A(T)}{1 + \exp\left(-\frac{u}{T}\right)}.$$

Нормувальна константа  $A(T)$  вводиться для умови нормалізації розподілу ймовірності [29]:

$$\int_0^1 \sigma(u) du = 1.$$

Параметр  $T$  визначає впорядкованість нейронної мережі. Якщо  $T$  спрямувати до 0, стохастичний нейрон перейде в звичайний нейрон з передавальною функцією Хевісайда (пороговою функцією), при будь-якому іншому значенні параметра  $T$  нейрон активується з ймовірністю  $P(u)$  [29].

Алгоритмом, що застосовується для тренування ОМБ, тобто для оптимізації вектора вагових коефіцієнтів, є алгоритм «контрастна дивергенція» (CD – Contrast Divergence). Цей алгоритм здійснює вибірку за Гіббсом, що використовується у процедурі градієнтного спуску для обчислення уточнених вагових коефіцієнтів.

При оптимізованих значеннях ваг мінімізується відстань Кульбака-Лейблера ( $D_{KL}$ ) між розподілом вхідних даних і наближенням моделі:

$$CD_n \propto D_{KL}(p_0(x)||p_\infty(x)) - D_{KL}(p_n(x)||p_\infty(x)). \quad (2.7)$$

,де  $p_n(x)$  є розподілом ланцюга Маркова, що виконується для  $n$  кроків . Навчання може бути виконано за допомогою одного кроку вибірки Гіббса. На цьому кроці виконуються операції: 1) оновлення ваг всіх прихованих нейронів, 2) оновлення ваг всіх видимих нейронів. Це відбувається згідно виразів [29]:

$$w_{ij}^{\tau+1} = w_{ij}^{\tau} + \alpha \left( \left( (h_j v_i)_{p(h|v;W)} \right)_0 - \left( h_j v_i \right)_n \right).$$

$$\left( (h_j v_i)_{p(h|v;W)} \right)_0 = \frac{1}{N} \sum_{k=1}^N v_i^{(k)} P \left( h_j^{(k)} = 1 | v^{(k)}; W \right).$$

$$\left( h_j v_i \right)_n = \frac{1}{N} \sum_{k=1}^N v_i^{(k)(n)} P \left( h_j^{(k)(n)} = 1 | v^{(k)(n)}; W \right).$$

де  $\tau$ -ітерація,  $\alpha$  - зміна коефіцієнтів ваг,  $N$  - загальна кількість вхідних екземплярів, верхній індекс  $(n)$  - кількість станів, отриманих після  $n$  ітерацій вибірки Гіббса з ланцюга Маркова, починаючи з  $P_0$  [29].

Рекурентні мережі є досить перспективними для задач прогнозування, а отже їх можна досить ефективно використовувати у системах керування. На сьогоднішній день є багато структур рекурентних мереж але серед них є одна з

найбільш вдалих саме для задач прогнозування числових рядів, ця структура NARX – "Нелінійна авторегресивна модель з екзогенним входом "

## 2.2. Рекурентна нейронна мережа NARX

NARX – це потужний клас моделей, який продемонстрував, що вони добре підходять для моделювання нелінійних систем та спеціальних часових рядів. Основне застосування динамічних нейронних мереж NARX – це передбачення функцій в системах управління. По суті про деякі важливі якості мереж NARX це алгоритм навчання: навчання ефективніше в мережах NARX, ніж в інших нейронних мережах (спуск градієнта краще в NARX) і ці мережі сходяться набагато швидше та оновлюються краще, ніж інші мережі [30].

Представлення простору станів рекурентних нейронних мереж NARX можна виразити як:

$$z_k(k+1) = \begin{cases} F(u(k), z_i(k)), & i=1 \\ z_i(k), & i=2,3,\dots,N \end{cases},$$

де вихід  $y(k) = z_i(k)$ , а  $z_i, i=1,2,\dots,N$  головна змінна рекурентної нейронної мережі, а  $u(k)$  попередні змінні,  $k$  величина часу.

Рекурентна мережа демонструє запам'ятовувальні властивості:

$$\lim_{m \rightarrow \infty} \frac{\partial z_i(k)}{\partial z_j(k-m)} = 0 \quad \forall k, m \in K, i \in O, j \in I,$$

де  $z$  головна змінна,  $m$  число кроків, "I" означає вхідних нейронів, "O" означає набір вихідних нейронів, "K" означає набір часових індексів [30].

Модель NARX, що використовується для наближення функції може бути реалізована багатьма способами, але найбільш просто здається, використати однонаправлену нейронну мережу з вбудованою пам'яттю (першою накладеною лінією затримки), як показано на рис. 2.1, плюс зворотній зв'язок – з'єднання від виходу другого шару до входу першого (другою накладеною лінією затримки).



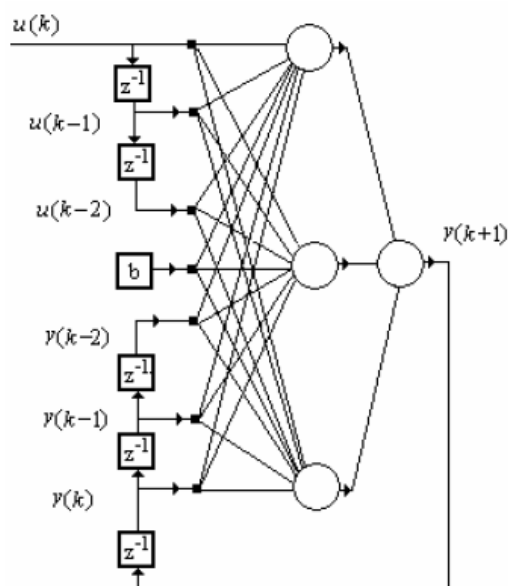


Рис. 2.1 Модель NARX

Зробити мережу залежною від  $d_u$  попередніх елементів послідовності ідентично до використання  $d_u$  входних блоків, що подаються  $d_u$  суміжними елементами послідовності. Цей вхід зазвичай називають часовим вікном, оскільки він забезпечує обмежений перегляд частини серії. Це також можна розглядати як простий спосіб перетворення тимчасового виміру в інший просторовий вимір [30].

На практиці було помічено, що прогнозування часових рядів буде покращено за рахунок одночасного аналізу відповідних часових рядів. Наприклад, споживання електроенергії на наступний день буде краще прогнозуватися, якщо взяти їх разом, з  $p_c$  добовими даними споживання та даними  $p_t$  температури навколишнього середовища одночасно, і застосувати їх як входи нейронної мережі. Архітектурна модель на рис. 2.2 зроблена для перевірки цієї гіпотези. Узагальнена реалізація цієї моделі дозволяє вводити та виводити багатовимірність і, таким чином, застосовувати "багатофакторний" часовий ряд.

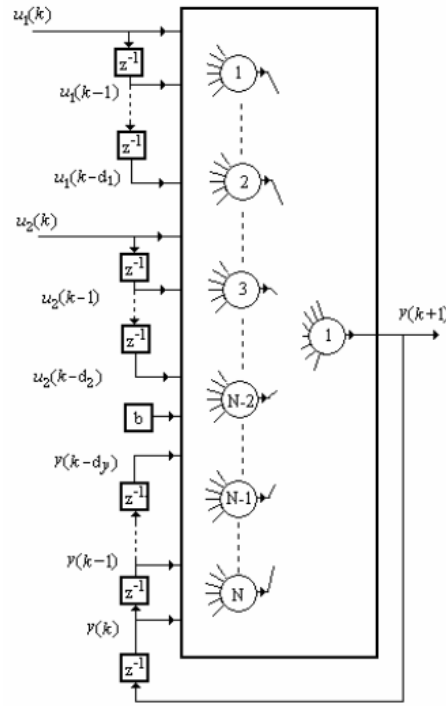


Рис. 2.2. Модель NARX з двома відведеними лініями затримки для двох часових рядів, застосованих у вході

Для архітектурної моделі на рис. 2.1 використовується позначення ШНМ  $(du, dy; N)$ , де  $du$  затримка введення,  $dy$  затримками виведення з  $N$  нейронами в шарі 1. Аналогічно, для архітектурної моделі на рис.6 використовується позначення  $(du1, du2, dy; N)$ .

Для моделей ШНМ, що використовуються в роботі, з двома рівнями (рівень 1 або вхідний шар та рівень 2 або вихідний шар) загальні рівняння прогнозування для обчислення наступного значення часового ряду  $y(k + 1)$  (вихід) з використанням моделі в рис. 2.1 , минуле спостереження  $u(k), u(k-1), \dots, u(k-du)$  та минулі результати  $y(k), y(k-1), \dots, y(k-dy)$  як вхідні дані можуть бути записані у формі:

$$y(k+1) = F_0 \left\{ w_{b0} + \sum_{h=1}^N w_{h0} * F_h(w_{h0} + \sum_{i=0}^{d_u} w_{ih} u(k-i)) + \sum_{j=0}^{d_y} w_{jh} y(k-j) \right\}. \quad (2.8)$$

Для моделі на рис. 2.2 рівняння прогнозування для обчислення вихідного значення  $y(k + 1)$  з використанням минулих спостережень  $u1(k), u2(k-1), \dots, u(k-du)$  для першого часового ряду, минулі спостереження  $u2(k), u(k-1), \dots, u2(k-du)$

для другого часового ряду та минулі виходи  $y(k), y(k-1), \dots, y(k-d_y)$  як вхідні дані можуть бути записані у формі [31]:

$$y(k+1) = F_0 \left\{ w_{b0} + \sum_{h=1}^N w_{h0} * F_h(w_{h0} + \sum_{i1=0}^{d_{u1}} w_{i1h} u_1(k-i_1)) + \sum_{i2=0}^{d_{u2}} w_{i2h} u_2(k-i_2) + \sum_{j=0}^{d_y} w_{jh} y(k-j) \right\}. \quad (2.9)$$

Для цілей навчання для обчислення градієнтів потрібен динамічний алгоритм розповсюдження зворотного зв'язку, який є більш обчислювально інтенсивним, ніж статичне зворотне розповсюдження і займає більше часу. Крім того, поверхні помилок для динамічних мереж можуть бути складнішими, ніж для статичних мереж. Навчання, швидше за все, потрапить у місцеві мінімуми. Вибраний метод тренінгу в цій роботі використовує перевагу наявності в тренувальному часі справжнього реального набору результатів. Можна використовувати справжній вихід замість розрахункового виходу для тренування мережі, яка має мережу з'єднань зворотного зв'язку, у яких з'єднання зворотного зв'язку роз'єднані (вирізані). Розв'язана мережа має загальну архітектуру подачі, яку можна навчити за допомогою класичного алгоритму статичного зворотного поширення. Крім того, під час тренінгу входи у мережу подачі інформації є лише справжніми / справжніми - не оціненими, і навчальний процес буде більш точним. Тренувальний процес має певні труднощі. Оскільки з'єднань або ваг міститься в мережі. Зазвичай ця кількість велика, і існує реальна небезпека "перетренувати" дані та створити помилкову відповідність, що не призводить до кращих прогнозів. Для моделі нейронної мережі NARX задається номером  $p = (d_u + d_y + 2) N$ . Рішення карає збільшення параметрів. Цей факт мотивує використання алгоритму, що включає техніку регуляризації, що передбачає зміну функції виконання для зменшення значення параметрів. Практично типова функціональна функція, що використовується у навчанні, MSE, замінюється новою, MSereg, таким чином:

$$MSE = \frac{1}{N} \sum_{i=1}^N (e_i)^2 = \frac{1}{N} \sum_{i=1}^N (t_i - y_i)^2,$$

$$MSE = \frac{1}{N} \sum_{i=1}^N \left( eMSW = \frac{1}{n} \sum_{j=1}^n w_{j,i}^2 \right)^2 = \frac{1}{N} \sum_{i=1}^N (t_i - y_i)^2,$$

$$MSE_{reg} = \xi MSE + (1 - \xi) MSW,$$

де  $t_i$  - ціль, а  $\xi$  - коефіцієнт продуктивності. Нова функціональна здатність призводить до того, що мережа має меншу вагу та ухили, і таким чином змушує реакцію мережі бути більш плавною та менш імовірною.

Мережева тренувальна функція, яка оновлює значення ваг та зміщення відповідно до оптимізації Левенберга-Маркарда, була модифікована для включення методики регуляризації. Вона мінімізує поєднання квадратних помилок і ваг, а потім визначає правильну комбінацію, щоб створити мережу, яка добре узагальнює. Процес називається байєсівською регуляцією.

Загалом, у задачах наближення функцій для мереж, що містять до декількох сотень ваг, алгоритм Левенберга-Маркварда матиме найшвидше зближення. Ця перевага особливо помітна, якщо потрібна дуже точна підготовка. Однак у міру збільшення кількості ваг у цій мережі перевага цього алгоритму зменшується.

## **Висновки до розділу 2**

1. Було представлено дві моделі рекурентних нейронних мереж «Обмежену машину Больцмана» та «Нелінійну авторегресивну модель з екзогенним входом».

2. Було проаналізовано моделі, висвітлено математичний апарат, описано систему навчання.

## РОЗДІЛ 3. МОДЕЛЮВАННЯ НЕЙРОНИХ МЕРЕЖ

### 3.1. База даних часових рядів генерації електроенергії

Центр «Спостереження, впливи, енергія» (O.I.E.) - це спільна дослідницька група MINES ParisTech – ARMINES [35], діяльність якої розташована на перехресті спостереження за енергією, довкіллям та землею. Вона вивчає та моделює "поновлювані джерела енергії" та впливи на навколишнє середовище, пов'язані з їх експлуатацією, спираючись на фундаментальні та прикладні наукові дисципліни (математика, метрологія, фізика, навколишнє середовище ...) та на технології інформація та комунікація. Веб-бази даних та сервіси є одним із головних механізмів поширення знань [35].

Служба SoDa бере свій початок від європейського проекту, який фінансувався Європейською Комісією з 1999 року. Проектом SoDa керувала група ParisTech.

SoDa впроваджується в 2003 році Mines ParisTech, а з 2013 року керується Науково-дослідним центром O.I.E. групи ParisTech (Центр спостереження, вплив, енергія), які доклали максимум зусиль для оцінки та аналізу просторових та часових аспектів відновлюваних джерел енергії та впливу енергетичних систем на навколишнє середовище [35].

Головний спосіб збору – даних ретроспективний аналіз Modern-Era для дослідження запропонованої версії 2 (MERRA-2) - це атмосферний програма НАСА для поширення експериментів з використанням нових моделей системи спостереження Землі Годдард версія 5 (GEOS-5) з її роботою асиміляції атмосферних даних (ADAS), версії 5.12.4. Проект MERRA фокусується на історичному аналізі клімату для широкого діапазону погодних і кліматичних тимчасових карт і вміщує набір навігацій НАСА EOS в кліматичному контексті.

MERRA-2 був ініційований як проміжний проект між старішими даними MERRA та наступним поколінням системного аналізу Землі, передбаченого для майбутнього поєднаного аналізу. Не маючи значних інвестицій для оновлення процедур засвоєння даних MERRA, системі бракувало можливості аналізу

останніх спостережень. Крім того, після заморожування системи MERRA в 2008 році було здійснено численні досягнення системи GEOS5. Тому було здійснено нову повну інтеграційну реалізацію. MERRA-2 охоплює період 1980-х років по теперішній час, продовжуючи як постійний аналіз клімату, коли ресурси дозволяють [35]. В табл. 3.1 наведено основні параметри даних що входять до бази даних.

Таблиця 3.1

Основні параметри даних що входять до бази даних.

# Site latitude (Широта)	44.083
# Site longitude (Довгота)	5.059
# Elevation (m) (Висота)	97
# Date beginning (Дата початку)	01.01.2017
# Date end (Дата кінця)	30.11.2017
# Tilt angle (degrees) (Кут нахилу)	25.0
# Azimuth angle (degrees) (Азимут нахилу)	180.0
# Albedo of the ground (Альбедо землі)	0.20
# Clear-Sky (Wh/m2) (Чисте небо)	Irradiation over the period if the sky were clear
# Top of Atmosphere (Wh/m2) (Стратосфера)	Irradiation over the period at the top of the atmosphere (extraterrestrial)
# Code (Кодування даних)	0: no data, 1: sun below horizon, 2: satellite assessment, 5: interpolation in time, 6: forecast
# Temperature (K) (Температура)	Temperature at 2 m above ground
# Relative humidity (%) (Відносна вологість)	Relative humidity at 2 m above ground
# Pressure (hPa) (Тиск)	Pressure at ground level
# Rainfall (kg/m2) (Дош)	Rainfall (= rain depth in mm)
# Snowfall (kg/m2) (Снігопад)	Snowfall

Більш детальний приклад бази даних в додатку 1.

Найбільш практичним є представлення даних на графіку рис. 3.1.

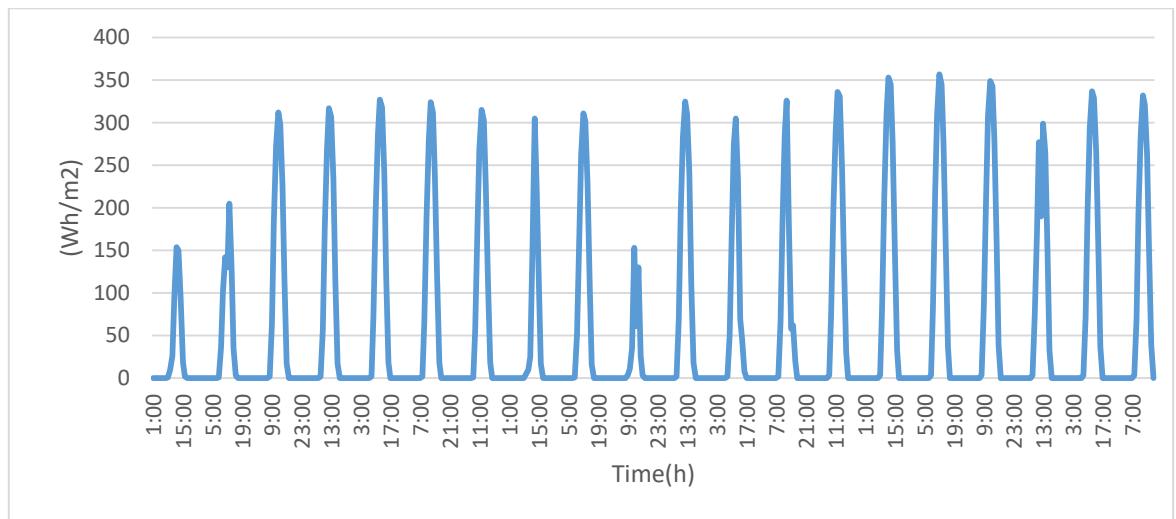


Рис. 3.1 Графік генерації електричної енергії від часу сонячної батареї в зимовий період

### 3.2. Моделювання в MATLAB

Програмне середовище MATLAB поєднує середовище моделювання, налаштоване на ітеративний аналіз та процеси проектування, з мовою програмування, яка безпосередньо виражає математику матриці та масиву.

Код MATLAB може бути надійно інтегрований з ІТ-системами підприємств, джерелами даних та операційними системами. MATLAB може співпрацювати з інженерними командами для:

- Забезпечення надійного виконання програми для Windows та Linux, як у власника, так і в хмарних мережах, таких як AWS та Microsoft Azure.
- Використовує стандартні галузеві механізми захисту для автентифікації, надання доступу та шифрування даних.
- Інтеграція безпосередньо в існуючі системи та дані, включаючи сучасні аналітичні системи, такі як Tableau, TIBCO Spotfire та Power BI.
- Використовує попередньо вбудовані, спеціалізовані для галузі бібліотеки MATLAB та Simulink, тому користувачі можуть швидко розпочати роботу [34].

По-перше використовуючи математичний апарат з розділу 2, а також інструментарій функцій матлабу була створена модель NARX (рис 3.2).

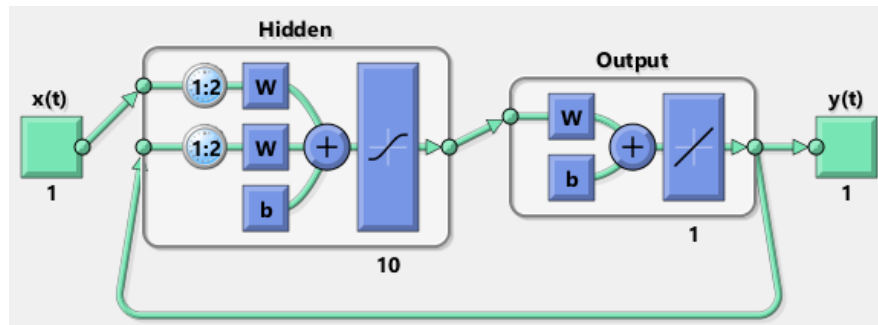


Рис. 3.2 Модель NARX в функції MATLAB

На рис. 3.2 показано  $x(t)$  вхідний блок, що посилає дані на Hidden-прихований шар, тут нейронна мережа оновлює початкові значення вагових коефіцієнтів та підбирає окремі значення нейронів.  $W$  та  $b$  це набір даних вагових коефіцієнтів. Цифра 10 означає що загалом при моделюванні використовувалась ШНМ з 10 нейронами. Блок Output (вихід) є нейроном, на виході цього блоку дані йдуть на вихід  $y(t)$ , а також в якості зворотнього зв'язку за попередньо обраним математичним законом оновлюються всі вагові коефіцієнти прихованого шару.

На рис. 3.3 зображено модель нейронної мережі, алгоритм навчання (Левенберга-Маркарда), характеристики навчального процесу нейронної мережі (кількість епох навчання, час, точність(mse), градієнт, кількість перевірок валідації).

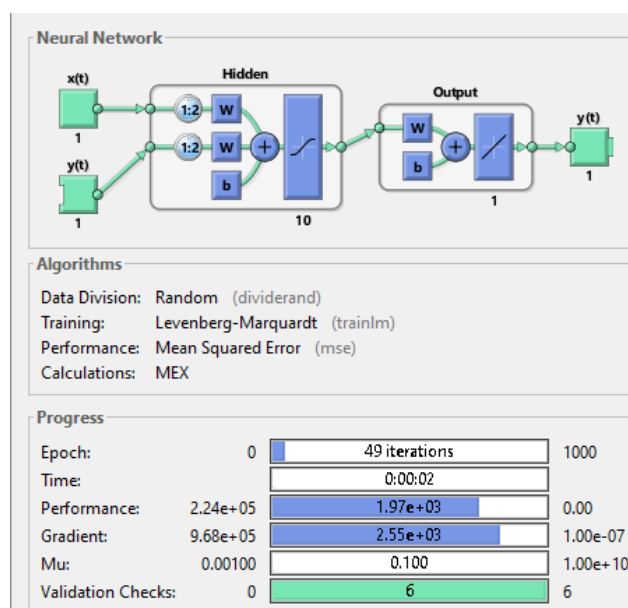


Рис. 3.3 Функціональна таблиця керування нейронною мережею



В результаті навчання мережі було отримано графік вихідного числового ряду і розподіл помилок в ряді (рис 3.4). Синім кольором виступають тренувальні мішені, зеленим валідаційні, червоним тестувальні. При навчанні використовувалась БД MINES ParisTech – ARMINES з кількістю значень 8000, показників генерації, температури та вологості.

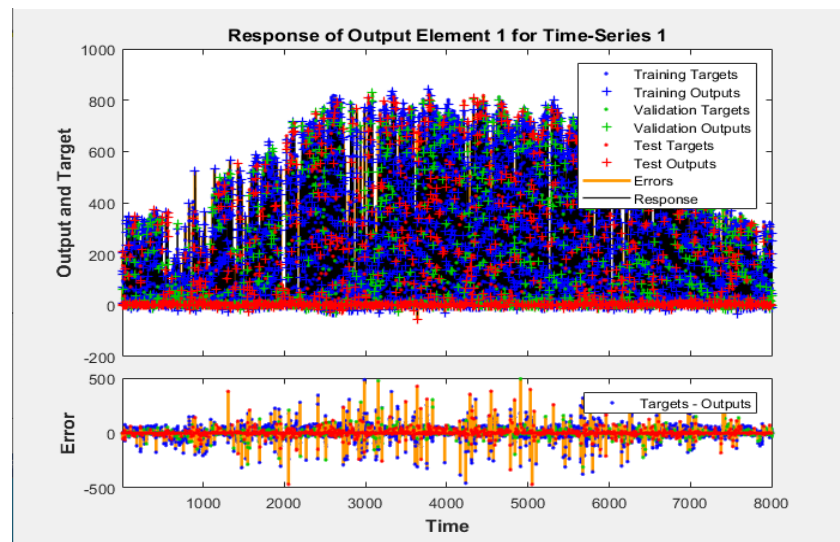


Рис. 3.4 Графік вихідного числового ряду а також розподіл помилок

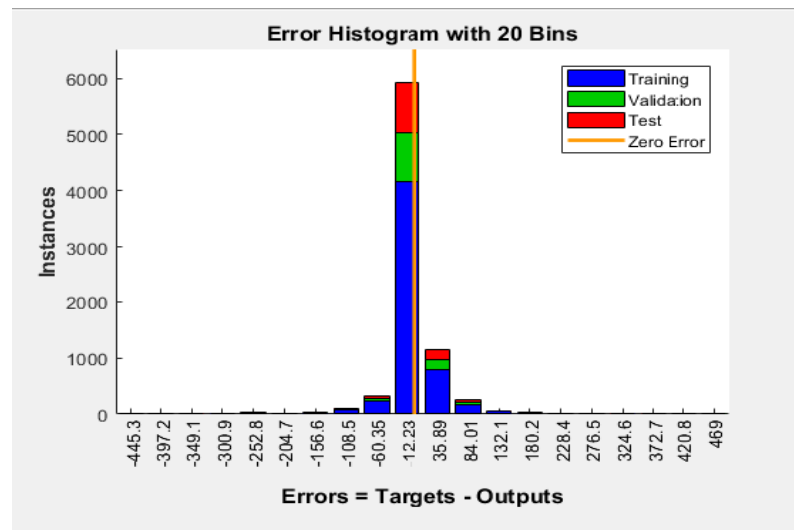


Рис. 3.5 Гістограма розподілу помилок з рис. 3.4

Згідно гістограми розподілу помилок (рис. 3.5) Більшість помилок знаходить на рівні 12% проте якщо брати до уваги і інші загально похибка передбачення складає ~20%.

Після тренування було пропущено ще 500 значень через мережу та порівняно з реальними даними (рис.3.6). Після було проведено експеримент для ще 500 значень данні яких було накладено для наглядності один на одний (рис. 3.7).

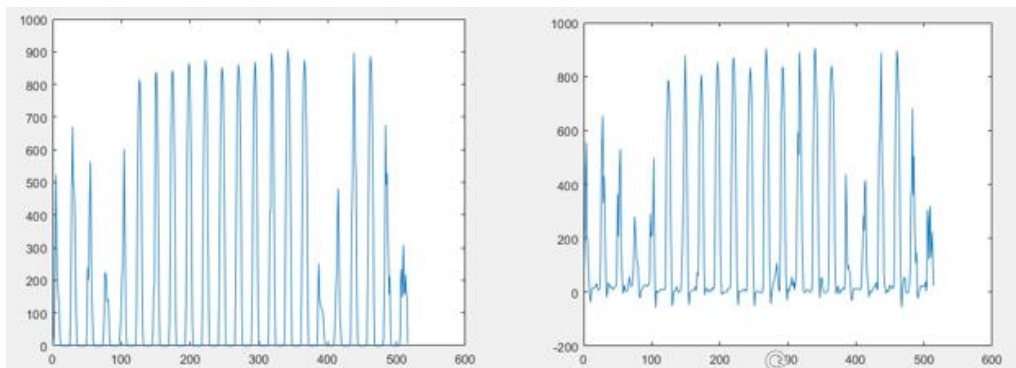


Рис. 3.6 Часові ряди електричної генерації змодельовані в середовищі MATLAB зліва реальний справа прогнозований



Рис. 3.7 Графік реальних та прогнозованих даних накладених один на одний

По-друге, була реалізована штучна нейронна мережа ОМБ, використовуючи нейроно-мережевий інструментарій в MATLAB і налаштування за замовчуванням (тобто кількість прихованих нейронів встановлено на 10 і 1 вихідний нейрон). Під час тренування параметри ШНМ (тобто ваги між нейронами та зміщеннями функціонування мережі) було використано алгоритм оптимізації Левенберга-Маркарда [34]. Та отримані результати навчання 1000 значень на рис. 3.8, похибка при цьому експерименті також була ~20%.

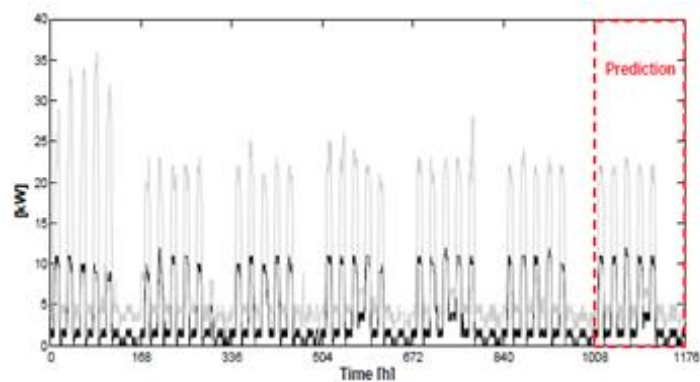


Рис. 3.8 Графік генерації електричної енергії з передбаченням ОМБ

### 3.3. Аналіз результатів моделювання

Якщо порівнювати обидві моделі на однакових даних отримуємо результати показані на рис. 3.9.

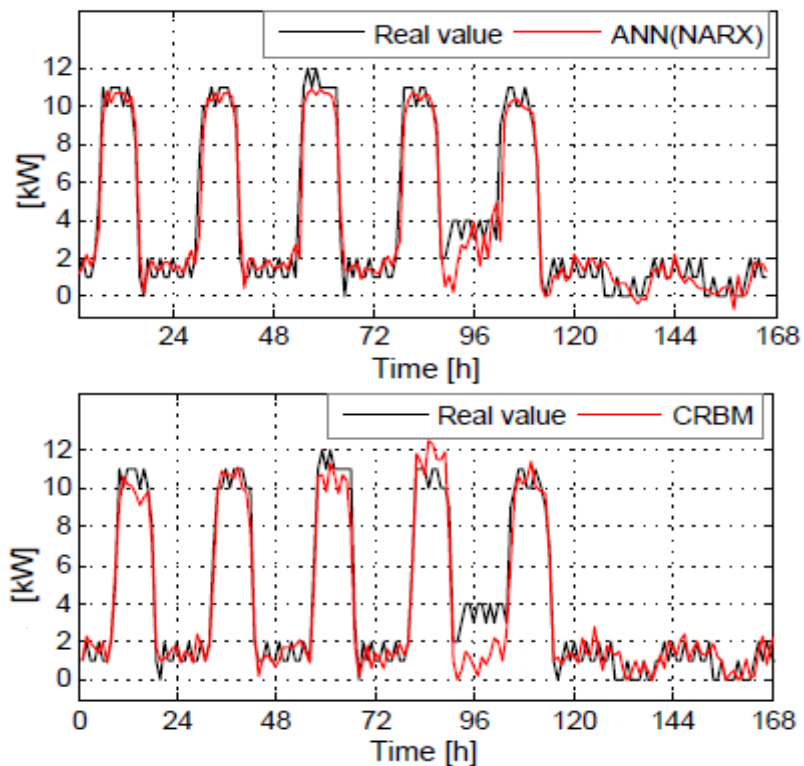


Рис. 3.9 Порівняння прогнозування часових рядів NARX та CRBM

Для того, щоб охарактеризувати точність моделей, було обчислено дві метрики. По-перше, точність прогнозування - це здатність метрики прогнозувати

з мінімальною середньою помилкою і може бути оцінена за кореневою середньою квадратичною помилкою (RMSE –Root Mean Square Error):

$$RMSE = \sqrt{(1/(T * n_v)) \sum_{(t=1)}^T \sum_{(i=1)}^{(n_v)} (v_{(i,t)} - v_{(i,t)}^*)^2},$$

де  $T$  являє собою загальну кількість кроків передбачування,  $v_i$  являє собою реальні значення для етапу часу  $t$ ,  $v_i^*$  являє собою передбачене значення.

По-друге, коефіцієнт кореляції Пірсона (PCC – Pearson correlation coefficient) [40] що вказує на ступінь лінійної залежності між реальною величиною та передбачуваним значенням:

$$PCC(v, v^*) = \frac{E[(v - \mu_v)(v^* - \mu_{v^*})]}{\sigma_v \sigma_{v^*}},$$

де  $\sigma_v$  та  $\sigma_{v^*}$  визначаються як стандартне відхилення  $v$  та  $v^*$ ,  $\mu_v$  - це середнє значення  $v$ ,  $\mu_{v^*}$  - це середнє значення  $v^*$ ,  $E$  - це очікування. Коефіцієнт кореляції може приймати будь-яке значення в межах діапазону  $[-1, 1]$ .

Таблиця 3.5

Порівняння характеристик точності методів NARX та CRBM

Метод прогнозування	RMSE	PCC
NARX	2.24	0,96
CRBM	7,04	0,85

Як бачимо показник RMSE моделі NARX значно краще, а PCC лише трішки краще.

### Висновки до розділу 3

1. Була описана база даних що використовувалась при моделюванні.
2. Було представлено моделювання різних методів прогнозування генерації електроенергії сонячними панелями в середовищі MATLAB.
3. Було експериментальним шляхом перевірено здатність прогнозування моделей NARX та CRBM.

Проведений аналіз показав, що NARX є кращим методом ніж CRBM.

## **РОЗДІЛ 4. РОЗРОБКА СТАРТАП ПРОЕКТУ**

Творець методики розвитку клієнтів (англ. Customer development) американський підприємець Стівен Бланк визначив стартапи як тимчасові структури, що існують для пошуку гнучкої і масштабованої бізнес-моделі [40]. Автор книги «Економічний стартап» і ідеолог інтерактивного підходу в підприємстві Ерік Рис зазначає, що стартапом може бути названа організація, що створює новий продукт або послугу (продуктову інновацію) в умовах високої невизначеності [38].

Стартап як форма малого ризикового (венчурного) підприємництва впродовж останнього десятиліття набула широкого розповсюдження у світі через зниження бар'єрів входу в ринок (із появою Інтернету як інструменту комунікацій та збуту стало простіше знаходити споживачів та інвесторів, займатись пошуком ресурсів, перетинати кордони між ринками різних країн), і вважається однією із наріжних складових інноваційної економіки, оскільки за рахунок мобільності, гнучкості та великої кількості стартап-проектів загальна маса інноваційних ідей зростає [36].

### **4.1.Опис ідеї проекту**

Метою розділу є формування інноваційного мислення, підприємницького духу та формування здатностей щодо оцінювання ринкових перспектив і можливостей комерціалізації основних науково-технічних розробок, сформованих у попередній частині магістерської дисертації у вигляді розроблення концепції стартап-проекту в умовах висококонкурентної ринкової економіки глобалізаційних процесів та має на меті проведення маркетингового аналізу стартап проекту задля визначення принципової можливості його ринкового впровадження та можливих напрямів реалізації цього впровадження [37]. Опис основної ідеї проекту описано в табл. 4.1.

## Опис ідеї проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Використати нейронну мережу NARX в якості модуля прогнозування генерації електроенергії відновлювальними джерелами в системі керування MicroGrid	1. Використання цієї системи в локальному MicroGrid	Підвищення енергоефективності системи MicroGrid
	2. Використання цієї системи в MicroGrid підключеної до глобальної мережі	Економія коштів при закупівлі електроенергії, можливість продати на ринку залишкову енергію

1. Аналіз потенційних техніко-економічних переваг ідеї (чим відрізняється від існуючих аналогів та замінників) порівняно із пропозиціями конкурентів.

На сьогоднішній день використання нейронних мереж сильно обмежено через ресурсомісткість подібних систем та великих необхідних технічних характеристик систем керування (найбільше об'єм пам'яті).

2. Визначення попереднього кола конкурентів (проектів-конкурентів):

Використання нейронних мереж не єдиний метод прогнозування. Існує багато методів прогнозування часових рядів такі як, Активно-адаптивна мережа (AAM), «AMIGO» (advanced microgrid optimization)...

3. Визначення сильних, слабких та нейтральних характеристик ідеї проекту.

Слабка сторона: великі об'єми даних для тренування, ресурсоємність.

Сильна сторона: простота використання натренованої мережі.

Нейтральна сторона: одноманітність аналізу при прогнозуванні.

4. Технологічний аудит ідеї проекту

Згідно ідеї проекту буде використано вже наявна технологія нейронної мережі NARX в сфері де вона не застосовувалась, а отже цілком можлива

технологічна реалізація проекту на будь-якій системі керування з достатнім ресурсом пам'яті.

## 5. Аналіз ринкових можливостей запуску стартаппроекту

Визначення ринкових можливостей, які можна використати під час ринкового впровадження проекту, та ринкових загроз, які можуть перешкодити реалізації проекту, дозволяє спланувати напрями розвитку проекту із урахуванням стану ринкового середовища, потреб потенційних клієнтів та пропозицій проектів-конкурентів.

На жаль в Україні немає шляхів реалізації даного проекту хоча б через те що тут немає ринку енергетичних ресурсів та вся енергетика монополізована компанією Укренерго.

В 2014 р. між Міністерством енергетики та вугільної промисловості і бельгійською компанією Tractebel було укладено контракт на розробку Техніко-економічного обґрунтування проекту. Запропоновано впровадження ряду пілотних технологій та проектів SmartGrid на рівні Системного Оператора[38].

Зокрема, *RES Forecasting*. Передбачено впровадження системи прогнозування виробітку генерації з ВДЕ. Ця задача повинна вирішуватись як самими учасниками для уникнення створення небалансів, так і системним оператором для оцінки нових проектів і оперативного планування енергорежимів. Наразі для консультацій з розробки тендерної документації на закупівлю програмно-технічних рішень проекту підписано контракт з компанією AF-Consult[39].

## 6. Ступеневий аналіз конкуренції на ринку

Таблиця 4.2

Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристик	Вплив на діяльність підприємства
1. Тип конкуренції : монополія	На ринку є тільки одна компанія	Конкуренція з національною

2. За рівнем конкурентної боротьби: національна	Компанія є національною	монопольною компанією в умовах
3. За галузевою ознакою: внутрішньогалузева	Боротьба в галузі енергетики	без відкритого ринку
4. За характером конкурентних переваг: нецінова	Енергоефективність є нецінковою перевагою	енергоресурсів неможлива

Прийнявши до уваги вищесказане робимо висновок що ринок України не є привабливим для входження.

## 7. SWOT- аналіз стартап-проекту

SWOT- аналіз стартап-проекту показаний в табл. 4.3.

Таблиця 4.3

SWOT- аналіз стартап-проекту

Сильні сторони: простота використання натренованої мережі	Слабкі сторони: великі об'єми даних для тренування, ресурсоемність
Можливості: підвищення енергоефективності	Загрози: Створення хибного прогнозу Зниження енергоефективності

Згідно данного аналізу можна сформулювати конкурентоспроможність данної системи.

## 8. Розроблення ринкової стратегії проекту

Стратегія збуту продукту та її основні клієнти:

- Опис профілю цільової групи потенційних клієнтів: компанії в сфері електроенергетики що виробляють електроенергію за допомогою ВДЕ.
- Готовність споживачів сприйняти продукт: ріст енергоефективності напряду дасть підвищення заробітку компанії.
- Інтенсивність конкуренції в сегменті: наразі подібні системи тільки почали виходити на ринок а отже конкуренція невелика.



- Компанія зосереджується на одному сегменті – тому обирає стратегію концентрованого маркетингу.

#### **Висновки до розділу 4**

Як висновок, основна ідея проекту використати відому технологію нейронної мережі у сфері енергетики для підвищення енергоефективності компаній, що виробляють електроенергію з ВДЕ. Серед сильних сторін простота використання системи, до слабких значна ресурсомісткість. Аналіз ринку України показує на неконкурентність даної системи через монополію національної компанії, зокрема відсутність вільного ринку енергоресурсів не дозволяє ефективно розпоряджатися власне створеною електроенергією. В таких умовах єдине рішення що можливе це співпраця лише з компанією Укренерго, або вихід на міжнародний ринок. Таке рішення значно зменшує конкурентноспроможність, але дозволяє представляти продукт більш ніж одному споживачу, а також є можливим на повну розкрити всі можливості даної системи через наявність відкритого ринку енергоресурсів.

## ВИСНОВКИ

1. MicroGrid — комплексна електротехнічна система, що складається з різних підсистем, які поєднані та взаємодіють між собою.
2. Штучна нейрона мережа – це набір штучних нейронів, з'єднаних між собою в деяку структуру і виконують певну задачу.
3. Проведений аналіз задачі прогнозування та використано багатокрокове з перенавчанням.
4. Наведено основні моделі нейронних мереж: персептрон Розенблата, одношарову та багатошарову прямого поширення, а також ШНМ Хопфілда.
5. Представлено дві моделі рекурентних нейронних мереж «Обмежену машину Больцмана» та «Нелінійну авторегресивну модель з екзогенним входом».
6. Проаналізовано моделі, висвітлено математичний апарат, описано систему навчання.
7. Описана база даних що використовувалась при моделюванні.
8. Представлено моделювання різних методів прогнозування генерації електроенергії сонячними панелями в середовищі MATLAB.
9. Було експериментальним шляхом перевірено здатність прогнозування моделей NARX та CRBM.
10. Проведений аналіз показав, що NARX є кращим методом ніж CRBM.
11. Було розроблено стартап-проект
12. Зроблено висновки з розробленого стартапу про співпрацю з НЕК «Укренерго», або вихід на міжнародний ринок збуту.

## СПИСОК ЛІТЕРАТУРИ

1. Михаил Шилер, Евгений Рубльовский, MicroGrid — ответ на новые вызовы электроэнергетики, CONTROL ENGINEERING РОССИЯ #4 (70), 2017
2. Стычинский З.А., Воропай Н.И. Возобновляемые источники энергии: теоретические основы, технологии, технические характеристики, экономика. – Иркутск 2010, – С. 28.
3. Кобец Б. Б.; Волкова И.О. Инновационное развитие электроэнергетики на базе концепции Smart Grid. – М.: ИАЦ Энергия, 2010. – С. 10-14.
4. Vehbi C. Gungor; Dilan Sahin; Taskin Kocak; Salih Ergut; Concettina Buccella; Carlo Cecati. «Smart Grid Technologies: Communication Technologies and Standards». IEEE Transactions On Industrial Informatics, Vol. 7, No. 4, November 2011
5. Bernd M. Buchholz, Zbigniew Styczynski. Smart Grids – Fundamentals and Technologies in Electricity Networks. – Magdeburg 2014
6. S. Denysiuk, V. Tarhonskyi, M. Artemiev, LOCAL ELECTRICAL ENERGY SYSTEMS WITH ACTIVE CONSUMER: METHODS OF CONSTRUCTION AND ALGORITHM OF THEIR FUNCTIONING, ISSN 1813-5420. Енергетика: економіка, технології, екологія. 2018. № 3
7. Fotis D. Kanellos, “Real-Time Control Based on Multi-Agent Systems for the Operation of Large Ports as Prosumer Microgrids,” IEEE Access, pp. 9439 – 9452, 19 May 2017.
8. Gen M., Cheng R., “Genetic Algorithms and Engineering Design,” John Wiley & Sons, №1, 1997.
9. Новотарский М.А. Имитационное моделирование нейросетей для решения уравнений математической физики локально-асинхронными методами // Радиоэлектроника. Информатика. Управление, 2001.–№1.– С.113-116.

10. Abbas H.M., Fahmy M.M. Neural model for Karhunen-Loeve transform with application to adaptive image compression // IEE Proceedings-I, 1993.–vol.140, №2.–P.135-143.
11. Amari S. Natural Gradient Works Efficiently in Learning // Neural Computation, 1998.–vol.10.–P.251-276.
12. Amari S., Cichocki A., Yang H.H. A new learning algorithm for blind signal separation // Advances in Neural Information Processing Systems, Touretzki D.S., Mozer M.C., Hasselmo M.E., Eds., MIT Press, 1996.–vol.8.–P.757-763.
13. Bäker M. Another look on multigrid // International Journal of Modern Physics C, 1997.–vol.8, №2.–P.191-205.
14. Baraldi A., Alpaydýn E. Simplified ART: A new class of ART algorithms // Int. Comput. Sci. Inst., Berkeley, CA, Tech. Rep. TR-98-004, 1998.–42 p.
15. Baraldi A., Parmiggiani F. A neural network for unsupervised categorization of multivalued input patterns: an application to satellite image clustering // IEEE Trans. Geosci. Remote Sensing, 1995.–vol.4, № 3.–P.393-396.
16. Bell A.J., Sejnowski T.J. An information-maximization approach to blind separation and blind deconvolution // Neural Computation, 1995.–vol.7, №6.–P.1029-1059.
17. Lanzkrom P.J., Rose D.J., Szyld D.B. Convergence of nested classical iterative methods for linear systems // Numerische Mathematik, 1991.–vol.58.–P.685-702.
18. Krishnapuram R., Keller J.M. A possibilistic approach to clustering // IEEE Trans. Fuzzy Syst., 1993.–vol.1.–P.98-110.
19. Kohonen T. Self-organizing maps: Optimization approaches // Proceedings of the International Conference on Artificial Neural Networks, Espoo, Finland, June 1991.–P. 981-990.
20. Kawamoto M., Barros A.K., Ohnishi N. A neural network for blind separation of convolved non-stationary signals // Proc. International ICSC Workshop on Independence and Artificial Neural Networks 98, Tenerife, Spain, 1998.–P.1374-1379.

21. Karnin E.D. A simple procedure for pruning back-propagation trained neural networks // IEEE transactions on neural networks, 1990.—vol.1, №2.—P.239-242.
22. Kalman R.E., Bucy R.S. New results in linear filtering and prediction. // Journal of Basic Engineering (ASME), 1961.— vol.83.—P.95-108.
23. Jankovic M.A New Modulated Hebbian Learning Rule – Method for Local Computation of a Principal Subspace // Proceedings of 8-th International Conference on Neural Information Processing, China, 2001.—6 p.
24. Haykin S. Neural Networks — A Comprehensive Foundation. Second edn, Prentice Hall Inc., Upper Saddle River, New Jersey, 1999.—842 p.
25. Hassibi B., Stork D., Wolff G.J. Optimal Brain Surgeon and General Network Pruning // IEEE Int. Conf. Neural Networks, San Francisco, 1993.—P.293-299.
26. Hackbush W. Multi-grid methods and applications. Springer series in computational mathematics 4, Berlin:SpringerVerlag, 1985.—542 p.
27. Miller K.D., MacKay D.J.C. The Role of Constraints in Hebbian Learning // Neural Computation, 1994.—vol.6.— P.100-126.
28. Ziehe A., Müller K.R. TDSEP – an efficient algorithm for blind separation using time structure // ICANN'98, Skovde, 1998.—P.675-680.
29. Zhyvoglyad O. S., *Prediction of electrical consumption network based on a neural network*, ISSN 2524-2725. ElectronAcoustEng, 2019, vol. 2, no.1.
30. Eugen Diaconescu, The use of NARX Neural Networks to predict Chaotic Time Series, WSEAS TRANSACTIONS on COMPUTER RESEARCH, Issue 3, Volume 3, March 2008.
31. M. T. Hagan, O. D. Jesus, R. Schultz, Training Recurrent Networks for Filtering and Control, in (editors) L.R. Medske, L.C. Jain, *Recurrent Neural Networks – Design and Applications*, CRC Press, 2001.
32. D.P. Mandic, J.A. Chambers, *Recurrent Neural Networks for Prediction*, JohnWiley&Sons, 2001.
33. Zina Boussaada, Octavian Curea, Ahmed Remaci, Haritza Camblong, Najiba Mrabet Bellaa, A Nonlinear Autoregressive Exogenous (NARX) Neural

Network Model for the Prediction of the Daily Direct Solar Radiation, *Energies* 2018, 11, 620

34. H.B. Demuth, M. Beale, *Users' Guide for the Neural Network Toolbox for Matlab*, The Mathworks, Natick, MA, 1998

35. Електронний ресурс HelioClim-3 is a satellite-derived solar radiation database <http://www.soda-pro.com/soda-products#time-series>

36. Mocanu, E. (2017). Machine learning applied to smart grids Eindhoven: Technische Universiteit Eindhoven

37. Методичні рекомендації до виконання розділу магістерських дисертацій для студентів інженерних спеціальностей / За заг. ред. О.А. Гавриша. – Київ : НТУУ «КПІ», 2016. – 28 с.

38. Електронний ресурс НЕК «Укренерго» <https://ua.energy/majbutnye-ukrenergo/smart-grid/>

39. Електронний ресурс Навчальні матеріали онлайн [https://pidruchniki.com/10981205/politekonomiya/vidi\\_konkurenciji](https://pidruchniki.com/10981205/politekonomiya/vidi_konkurenciji)

40. Електронний ресурс Вікіпедія Стартапи <https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D0%B0%D1%80%D1%82%D0%B0%D0%BF>

## ДОДАТОК 1

### Скрипт-код функції нейронної мережі NARX в системі для моделювання MATLAB

Код генерує математичний апарат нейронної мережі згідно Розділу 2, а також містить різні функціональні додатки для спрощеного керування мережею: Closed Loop Network, Multi-step Prediction, Step-Ahead Prediction Network.

```
//  
function [Y,Xf,Af] = myNeuralNetworkFunction(X,Xi,~)  
%MYNEURALNETWORKFUNCTION neural network simulation function.  
% Generated by Neural Network Toolbox function genFunction, 07-Dec-2019  
19:02:20.  
% [Y,Xf,Af] = myNeuralNetworkFunction(X,Xi,~) takes these arguments:  
% X = 2xTS cell, 2 inputs over TS timesteps  
% Each X{1,ts} = 1xQ matrix, input #1 at timestep ts.  
% Each X{2,ts} = 1xQ matrix, input #2 at timestep ts.  
%  
% Xi = 2x2 cell 2, initial 2 input delay states.  
% Each Xi{1,ts} = 1xQ matrix, initial states for input #1.  
% Each Xi{2,ts} = 1xQ matrix, initial states for input #2.  
%  
% Ai = 2x0 cell 2, initial 2 layer delay states.  
% Each Ai{1,ts} = 10xQ matrix, initial states for layer #1.  
% Each Ai{2,ts} = 1xQ matrix, initial states for layer #2.  
%  
% and returns:  
% Y = 1xTS cell of 2 outputs over TS timesteps.  
% Each Y{1,ts} = 1xQ matrix, output #1 at timestep ts.  
%  
% Xf = 2x2 cell 2, final 2 input delay states.  
% Each Xf{1,ts} = 1xQ matrix, final states for input #1.  
% Each Xf{2,ts} = 1xQ matrix, final states for input #2.  
%  
% Af = 2x0 cell 2, final 0 layer delay states.  
% Each Af{1ts} = 10xQ matrix, final states for layer #1.  
% Each Af{2ts} = 1xQ matrix, final states for layer #2.  
%  
% where Q is number of samples (or series) and TS is the number of timesteps.  
%#ok<*RPMT0>  
% ===== NEURAL NETWORK CONSTANTS =====  
% Input 1  
x1_step1.xoffset = 266.7;  
x1_step1.gain = 0.043965706748736;
```

```

x1_step1.ymin = -1;
% Input 2
x2_step1.xoffset = 0;
x2_step1.gain = 0.00239520958083832;
x2_step1.ymin = -1;
% Layer 1
b1 = [0.47593116421494507939;-
3.2462135737808774927;0.27224232774262235868;-
0.24062154652250639097;0.079812392972375995059;-
0.037804750905213256595;-3.0670295008656895952;0.43426559346534721051;-
2.7113105614342400962;-11.006055942476317355];
IW1_1 = [-9.9260136453647369592
9.8579979284351946234;1.9643108987502095886 -0.63316012696097079271;-
6.847080970948388412 7.008734239795342269;-1.5316374883691465936
2.183088027054325142;0.97589235266907692345 -1.5669547191981554324;-
1.463335467630241693 0.98927620204428146078;0.71404659921855584415 -
0.18144676507398382892;0.015360770993487198044
1.0516885069843020517;0.075408048913736508423
0.28568352426538734523;4.0707095346699038174 -4.0195081239096683845];
IW1_2 = [-0.34849748442210198851
0.47273498682417730565;0.36208047320748459885 -
1.981461136366994058;2.263365122357534176 -
2.2302943051619035231;2.4445622299547706824 -1.1824867343331719738;-
1.7492002128780921311 0.4882901652771122758;-0.76409128089319822141 -
0.55568218308183336962;-1.4310655205563780346 4.3056288993949376476;-
1.1984540317707030876 -0.51277848394615621608;-0.79426341478688156084
2.8621776687970568176;-8.1375409309987638551 -1.3184539884381560881];
% Layer 2
b2 = 0.11319996130426070169;
LW2_1 = [-0.41031601449711219143 -0.17990753870087308752
0.27194889283547951653 1.0872375991446365795 1.5335773678019806177 -
0.91649842971870287833 -1.8142999283351426421 -0.28974704400531359205
4.2388956771811576729 -2.1385984839531584356];
% Output 1
y1_step1.ymin = -1;
y1_step1.gain = 0.00239520958083832;
y1_step1.xoffset = 0;
% ===== SIMULATION =====
% Format Input Arguments
isCellX = iscell(X);
if ~isCellX
    X = {X};
end
if (nargin < 2), error('Initial input states Xi argument needed.');
```



```

% Dimensions
TS = size(X,2); % timesteps
if ~isempty(X)
    Q = size(X{1},2); % samples/series
elseif ~isempty(Xi)
    Q = size(Xi{1},2);
else
    Q = 0;
end
% Input 1 Delay States
Xd1 = cell(1,3);
for ts=1:2
    Xd1{ts} = mapminmax_apply(Xi{1,ts},x1_step1);
end
% Input 2 Delay States
Xd2 = cell(1,3);
for ts=1:2
    Xd2{ts} = mapminmax_apply(Xi{2,ts},x2_step1);
end
% Allocate Outputs
Y = cell(1,TS);
% Time loop
for ts=1:TS
    % Rotating delay state position
    xdts = mod(ts+1,3)+1;
    % Input 1
    Xd1{xdts} = mapminmax_apply(X{1,ts},x1_step1);
    % Input 2
    Xd2{xdts} = mapminmax_apply(X{2,ts},x2_step1);
    % Layer 1
    tapdelay1 = cat(1,Xd1{mod(xdts-[1 2]-1,3)+1});
    tapdelay2 = cat(1,Xd2{mod(xdts-[1 2]-1,3)+1});
    a1 = tansig_apply(repmat(b1,1,Q) + IW1_1*tapdelay1 + IW1_2*tapdelay2);
    % Layer 2
    a2 = repmat(b2,1,Q) + LW2_1*a1;
    % Output 1
    Y{1,ts} = mapminmax_reverse(a2,y1_step1);
end
% Final Delay States
finalxts = TS+(1:2);
xits = finalxts(finalxts<=2);
xits = finalxts(finalxts>2)-2;
Xf = [Xi(:,xits) X(:,xits)];
Af = cell(2,0);

```

```

% Format Output Arguments
if ~isCellX
    Y = cell2mat(Y);
end
end
% ===== MODULE FUNCTIONS =====
% Map Minimum and Maximum Input Processing Function
function y = mapminmax_apply(x,settings)
y = bsxfun(@minus,x,settings.xoffset);
y = bsxfun(@times,y,settings.gain);
y = bsxfun(@plus,y,settings.ymin);
end
% Sigmoid Symmetric Transfer Function
function a = tansig_apply(n,~)
a = 2 ./ (1 + exp(-2*n)) - 1;
end
% Map Minimum and Maximum Output Reverse-Processing Function
function x = mapminmax_reverse(y,settings)
x = bsxfun(@minus,y,settings.ymin);
x = bsxfun(@rdivide,x,settings.gain);
x = bsxfun(@plus,x,settings.xoffset);
end
% Solve an Autoregression Problem with External Input with a NARX Neural
Network
% Script generated by Neural Time Series app
% Created 07-Dec-2019 19:02:34
%
% This script assumes these variables are defined:
%
% temp - input time series.
% reg - feedback time series.

X = tonndata(temp,false,false);
T = tonndata(reg,false,false);
% Choose a Training Function
% For a list of all training functions type: help nntrain
% 'trainlm' is usually fastest.
% 'trainbr' takes longer but may be better for challenging problems.
% 'trainscg' uses less memory. Suitable in low memory situations.
trainFcn = 'trainlm'; % Levenberg-Marquardt backpropagation.
% Create a Nonlinear Autoregressive Network with External Input
inputDelays = 1:2;
feedbackDelays = 1:2;
hiddenLayerSize = 10;

```

```

net = narxnet(inputDelays,feedbackDelays,hiddenLayerSize,'open',trainFcn);
% Choose Input and Feedback Pre/Post-Processing Functions
% Settings for feedback input are automatically applied to feedback output
% For a list of all processing functions type: help nnprocess
% Customize input parameters at: net.inputs{i}.processParam
% Customize output parameters at: net.outputs{i}.processParam
net.inputs{1}.processFcns = {'removeconstantrows','mapminmax'};
net.inputs{2}.processFcns = {'removeconstantrows','mapminmax'};
% Prepare the Data for Training and Simulation
% The function PREPARETS prepares timeseries data for a particular network,
% shifting time by the minimum amount to fill input states and layer
% states. Using PREPARETS allows you to keep your original time series data
% unchanged, while easily customizing it for networks with differing
% numbers of delays, with open loop or closed loop feedback modes.
[x,xi,ai,t] = preparets(net,X,{ },T);
% Setup Division of Data for Training, Validation, Testing
% For a list of all data division functions type: help nndivision
net.divideFcn = 'dividerand'; % Divide data randomly
net.divideMode = 'time'; % Divide up every sample
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;
% Choose a Performance Function
% For a list of all performance functions type: help nnperformance
net.performFcn = 'mse'; % Mean Squared Error
% Choose Plot Functions
% For a list of all plot functions type: help nnplot
net.plotFcns = {'plotperform','plottrainstate', 'ploterrhist', ...
    'plotregression', 'plotresponse', 'ploterrcorr', 'plotinerrcorr'};
% Train the Network
[net,tr] = train(net,x,t,xi,ai);
% Test the Network
y = net(x,xi,ai);
e = gsubtract(t,y);
performance = perform(net,t,y)
% Recalculate Training, Validation and Test Performance
trainTargets = gmultiply(t,tr.trainMask);
valTargets = gmultiply(t,tr.valMask);
testTargets = gmultiply(t,tr.testMask);
trainPerformance = perform(net,trainTargets,y)
valPerformance = perform(net,valTargets,y)
testPerformance = perform(net,testTargets,y)
% View the Network
view(net)

```

```

% Plots
% Uncomment these lines to enable various plots.
% figure, plotperform(tr)
% figure, plottrainstate(tr)
% figure, ploterrhist(e)
% figure, plotregression(t,y)
% figure, plotresponse(t,y)
% figure, ploterrcorr(e)
% figure, plotinerrcorr(x,e)
% Closed Loop Network
% Use this network to do multi-step prediction.
% The function CLOSELOOP replaces the feedback input with a direct
% connection from the outout layer.
netc = closeloop(net);
netc.name = [net.name ' - Closed Loop'];
view(netc)
[xc,xic,aic,tc] = preparets(netc,X,{ },T);
yc = netc(xc,xic,aic);
closedLoopPerformance = perform(net,tc,yc)
% Multi-step Prediction
% Sometimes it is useful to simulate a network in open-loop form for as
% long as there is known output data, and then switch to closed-loop form
% to perform multistep prediction while providing only the external input.
% Here all but 5 timesteps of the input series and target series are used
% to simulate the network in open-loop form, taking advantage of the higher
% accuracy that providing the target series produces:
numTimesteps = size(x,2);
knownOutputTimesteps = 1:(numTimesteps-5);
predictOutputTimesteps = (numTimesteps-4):numTimesteps;
X1 = X(:,knownOutputTimesteps);
T1 = T(:,knownOutputTimesteps);
[x1,xio,aio] = preparets(net,X1,{ },T1);
[y1,xfo,afo] = net(x1,xio,aio);
% Next the the network and its final states will be converted to
% closed-loop form to make five predictions with only the five inputs
% provided.
x2 = X(1,predictOutputTimesteps);
[netc,xic,aic] = closeloop(net,xfo,afo);
[y2,xfc,afc] = netc(x2,xic,aic);
multiStepPerformance = perform(net,T(1,predictOutputTimesteps),y2)
% Alternate predictions can be made for different values of x2, or further
% predictions can be made by continuing simulation with additional external
% inputs and the last closed-loop states xfc and afc.

```

```

% Step-Ahead Prediction Network
% For some applications it helps to get the prediction a timestep early.
% The original network returns predicted  $y(t+1)$  at the same time it is
% given  $y(t+1)$ . For some applications such as decision making, it would
% help to have predicted  $y(t+1)$  once  $y(t)$  is available, but before the
% actual  $y(t+1)$  occurs. The network can be made to return its output a
% timestep early by removing one delay so that its minimal tap delay is now
% 0 instead of 1. The new network returns the same outputs as the original
% network, but outputs are shifted left one timestep.
nets = removedelay(net);
nets.name = [net.name ' - Predict One Step Ahead'];
view(nets)
[xs,xis,ais,ts] = preparets(nets,X,{ },T);
ys = nets(xs,xis,ais);
stepAheadPerformance = perform(nets,ts,ys)
% Deployment
% Change the (false) values to (true) to enable the following code blocks.
% See the help for each generation function for more information.
if (false)
    % Generate MATLAB function for neural network for application
    % deployment in MATLAB scripts or with MATLAB Compiler and Builder
    % tools, or simply to examine the calculations your trained neural
    % network performs.
    genFunction(net,'myNeuralNetworkFunction');
    y = myNeuralNetworkFunction(x,xi,ai);
end
if (false)
    % Generate a matrix-only MATLAB function for neural network code
    % generation with MATLAB Coder tools.
    genFunction(net,'myNeuralNetworkFunction','MatrixOnly','yes');
    x1 = cell2mat(x(1,:));
    x2 = cell2mat(x(2,:));
    xi1 = cell2mat(xi(1,:));
    xi2 = cell2mat(xi(2,:));
    y = myNeuralNetworkFunction(x1,x2,xi1,xi2);
end
if (false)
    % Generate a Simulink diagram for simulation or deployment with.
    % Simulink Coder tools.
    gensim(net);
end

```

## ABSTRACT

**Actuality of theme.** MicroGrid is a sophisticated electronic system that includes different subsystems that are interconnected and interconnected. This network is built from alternate and conventional energy pockets and later loads combined into one system. The main feature of MicroGrid is that, notwithstanding the functionality of the overall network framework, it is longer insulated due to the lower tariff rate and the highest energy efficiency. Alternative-powered systems are attractive because they receive subsidies and discounts for energy distribution, the so-called "green tariff", and are therefore economically viable. But alternative sources of electricity generate volatility due to weather, daily cycles, etc., so they need a more sophisticated control system.

When solving this problem, the question arises of forecasting power generation in individual subsystems. Given the large amount of data to be processed in real time, as well as the "human factor" that further complicates the task of predicting power generation in MicroGrid, the use of artificial neural networks is promising.

**The purpose and objectives of the study.** The aim of the master's thesis is to develop a method for predicting the generation volumes from alternative sources of electricity in the MicroGrid system based on artificial neural networks.

**This goal requires the following scientific and practical problems:**

1. Review of the existing structures of the distributed generation MicroGrid systems, analysis of their characteristics, formulation and justification of the problems of forecasting the generation of electricity from alternative sources.
2. Comparative analysis of forecasting methods, selection and justification of the artificial neural network method for the prediction task in MicroGrid.
3. Investigation of types and structures of artificial neural networks, their comparison and selection of the most appropriate structure for the task.
4. Development of a method for predicting power generation based on artificial neural network.

5. Simulation of work of artificial neural network, processing and analysis of results.

**The object of the study** is the processes of generating electricity from renewable and alternative sources within the MicroGrid distributed generation systems.

**The subject of the study** is the prediction of electricity generation in MicroGrid using a mathematical apparatus of artificial neural networks.

**Relationship with working with scientific programs, topics, plans.** The research work on the topic of the master's thesis was carried out in accordance with the directions of scientific research and the work plan of the department of industrial electronics KPI them. Igor Sikorsky.

**Research methods.** The mathematical apparatus of artificial neural networks with NARX architecture was used to solve prediction problems. To confirm the theoretical calculations, we used the MATLAB software environment, which implemented the proposed neural network structure.

**The scientific novelty of the obtained results is the following:**

- It was first proposed to use a mathematical apparatus of artificial neural networks to solve the problem of predicting electricity generation in MicroGrid;
- The comparative analysis of different structures of artificial neural networks is carried out, the choice of NARX architecture for the solution of the task is substantiated.

**The practical significance of the results obtained:**

- the use of artificial neural networks to predict the generation of electricity coming from alternative power sources in MicroGrid, allows you to operate with large amounts of data, which is extremely relevant for modern smart systems such as "smart home", in particular, with the implementation of the concept of "Internet of Things" ;
- The theoretical results of the work are confirmed by the results of modeling in the MATLAB software environment, which testifies to the feasibility and validity of the chosen architecture for forecasting.

The results of the work were tested in the framework of a scientific report at the XII International Scientific and Technical Conference of Young Scientists "Electronics-2019", April 23-25, 2019.

**Publications.** The main provisions and results of the master's study are covered in the following publications:

1. View of OS Prediction of neural network power consumption. // Electronic and Acoustic Engineering - 2019. Volume 2, No.1. - P. 39-42. ISSN 2524-2725.

2. Zhyvoglyad O.S. System of automatic control of power usage based on a neural network "- article submitted for publication in the professional periodical" Lighting and Electric Power Engineering " -that-electricity. The article was successfully reviewed on November 18, 2019 and is due for publication in the forthcoming issue of the journal.

**Structure and scope of work.** The master's thesis consists of 4 sections, general conclusions, a list of literature of 40 titles and 1 appendices. Each section has its own conclusions. The total volume of work is 80 pages.